

## Using OpenOffice as a Portable Interface to JAVA-Based Applications

Tom Comeau, Bruce Garrett, Joel Richon, Fred Romelfanger

*Space Telescope Science Institute*

**Abstract.** STScI previously used Microsoft Word and Microsoft Access, a Sybase ODBC driver, and the Adobe Acrobat PDF writer, along with a substantial amount of Visual Basic, to generate a variety of documents for the internal Space Telescope Grants Administration System (STGMS). While investigating an upgrade to Microsoft Office XP, we began considering alternatives, ultimately selecting an open source product, OpenOffice.org. This reduces the total number of products required to operate the internal STGMS system, simplifies the build system, and opens the possibility of moving to a non-Windows platform. We describe the experience of moving from Microsoft Office to OpenOffice.org, and our other internal uses of OpenOffice.org in our development environment.

### 1. Introduction

The Space Telescope Grants Management System (STGMS) is an integrated set of applications and web services used by both internal and external users to manage HST-related grants.

The external STGMS application is a downloadable Java application used by HST investigators and sponsored programs offices. It allows grantees to prepare budgets, manage program funding, and submit performance and financial reports. The external application is written entirely in Java, and runs on a variety of platforms.

The internal STGMS application is used to administer HST grants. It is used only by the Grants Administration and Finance Offices at STScI. One key feature of STGMS is the ability to generate PDF documents for grant awards and modifications, and store those documents in a database. External users can then retrieve their own documents using the external application.

### 2. The Original System

The original system used Microsoft Office to prepare documents, convert and store documents. The Microsoft Office interface was extremely complicated, relying on Java, VBA, COM objects, two Office components, a Sybase ODBC driver, and a separate Adobe PDF Writer.

To create a document, the internal STGMS application invoked a VB program which launched Microsoft Word and Microsoft Access, Access extracted the required data from a Sybase database using ODBC into an Access table. The

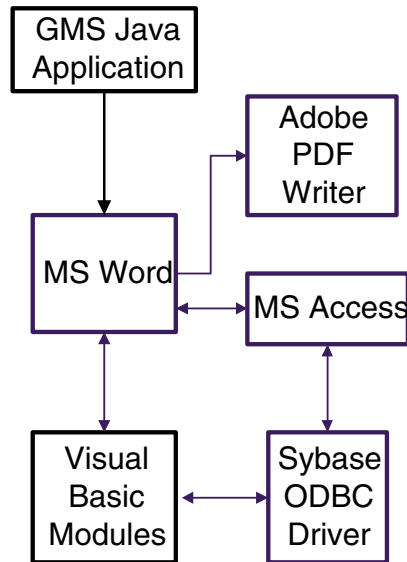


Figure 1. The Microsoft Office solution required multiple interfaces, and worked only on Windows platforms.

VB application then connected the Access data source to Word document and allowed the user to make any required changes before submitting the document. Word then printed the document to a PDFWriter, and saved the document to the database using ODBC. If the recipient of the document wanted an electronic copy mailed, the text of the Word document was extracted and emailed using VBA. An outline of the interfaces is shown in Figure 1.

The entire process was complicated and prone to configuration errors. Because the Word documents had embedded macros, internal users were not allowed to make permanent changes. All changes required developer support, which slowed document changes and increased the cost of changes.

### 3. Abandoning Microsoft

While investigating a possible upgrade to Office XP, we also began investigating alternatives and quickly settled on OpenOffice.org. OpenOffice.org 1.1 was officially released October 1st. Integration testing of OpenOffice.org with STGMS is underway now.

STGMS uses the Universal Network Objects (UNO) interface to start the OpenOffice.org Writer and give the Writer access to STGMS data. All of the data access functions have been moved out of VB and into the Java application. The OpenOffice.org documents have been made as simple as possible, and significant parts of the existing Java code were reused to provide the data needed by the OpenOffice.org documents. The simplified interfaces are shown in Figure 2.

Once started, the OpenOffice.org writer uses STGMS classes to fill in the user variables (fields) The application tells the Writer to export PDF to a file, and the application saves the file using JDBC.



Figure 2. The OpenOffice.org solution is significantly simpler and works across platforms.

OpenOffice.org has an export to PDF facility that eliminates the need for the Adobe PDF writer. This facility is also used by developers to exchange documents across platforms.

Unlike the Microsoft Office implementation, our OpenOffice.org implementation uses no embedded macros. Because the document templates are macro-free, they can be edited by Grants Administrator, rather than developers. This significantly lowers the cost of changes and the time required to make permanent changes to the templates.

#### 4. OpenOffice.org Challenges

From a developer perspective, it is still challenging to understand and use the OpenOffice.org API. JDBC support currently does not work correctly with some databases, in particular the Sybase JDBC driver. Access via JDBC would have been useful, but was not essential for our project. Instead we use JDBC from the Java application to extract data and pass it to OpenOffice.org documents.

OpenOffice.org has a bean interface, but that interface is not supported, and requires that we write native code. A working bean interface that did not require native code would have been extremely helpful. Instead, we have implemented changes to the OpenOffice.org menus, substituting our own menu item for the OpenOffice.org file menu. The Java application listens for our custom menu item and takes control back from the OpenOffice.org Writer.

The API is not easy to learn. The class hierarchy is not obvious, and the Java and Python interfaces are clearly add-ons to the underlying C++ application.

OpenOffice.org does have a VB-like language, but we are not using that facility. We are explicitly reducing the number of languages used in order to simplify STGMS maintenance.

#### 5. Conclusion

OpenOffice.org reduces the total number of components needed and provides the option to move off the Microsoft Windows platform. Because OpenOffice.org works very well on Windows, we are not moving off Windows at this time.

From a user perspective, OpenOffice.org does most to all of what Microsoft Office does at a much lower cost. Because OpenOffice.org is Open Source, and STGMS is able to use simpler interfaces, we expect the cost to maintain STGMS with OpenOffice.org will be significantly lower.