ABSTRACT
With the release of the SAS BI Server in 2003, SAS provided a suite of tools that made the power of SAS available to everyone in an organization. As more organizations deploy these tools, questions are being asked about managing the content that’s generated with these tools in a dynamic environment. How can we move a report from the test environment to the production environment? How can we turn an ad-hoc report into a batch report? How can I ensure that my report gets to the right people? This paper attempts to answer these questions by describing which tools and techniques are available for dealing with distribution, automation, promotion, security, and archiving reports in the SAS BI environment.

INTRODUCTION
The SAS BI Server enables an organization to extend the power of SAS software to everyone in the organization, regardless of their programming skills or knowledge about computers. Using a set of interfaces that are tailored to specific user communities and skill sets, you can perform ad-hoc query and reporting, OLAP data exploration, and data analysis. Most of the interfaces are Web-based and accessed via a browser; others are seamlessly integrated into the Microsoft Office products. All are designed to enable users to focus on answering business questions and solving business problems instead of learning how to program or use a new tool.

Although the interfaces and their use for creating reports are important, a successful BI implementation also involves administrative activities that manage the BI content that’s produced by users of these interfaces. This paper focuses on some of these activities, specifically, content distribution, automation, promotion, archiving, security, and how organizations might handle these issues when using the SAS BI Server.

WHAT IS BUSINESS INTELLIGENCE CONTENT?
The SAS BI Server is composed of a number of interfaces that produce some type of content. Before discussing how to manage this content, it might be helpful to summarize the interfaces and the types of content that each interface produces or surfaces. Given the central role that SAS Web Report Studio plays in the SAS BI Server, a major part of this paper focuses on how to manage the content created by using this interface.

SAS Web Report Studio – is a Web-based end-user reporting interface that provides an easy-to-use tool for creating reports. In addition to building reports based on the Information Maps (which are created using SAS Information Map Studio), report content can be generated by a Stored Process or imported from other SAS interfaces such as SAS Web OLAP Viewer for Java, SAS Enterprise Guide, or Base SAS. The types of BI content that are produced are SAS Reports and Report templates.

SAS Information Map Studio – an interface that enables data architects to create information maps. Information maps consist of a layer of metadata that hides the details of an organization’s data model but permits end-users to work with the business concepts and terminology that they’re familiar with. The type of BI content that is produced is Information Maps.

SAS Stored Processes Authoring tools – A SAS Stored Process is a unit of SAS programming logic that has been packaged in a way that makes it usable by non-programmers, and it can be shared across the enterprise. Stored processes can create information as well as reports. In addition to the SAS code, a stored process contains metadata that describes any parameters that can be set to control the output that’s created and the environment in which the code should be run. Because a stored process is primarily SAS code, any of the tools that you currently use to author SAS code can be used to author the stored process code. To register the stored process metadata, you can use SAS Enterprise Guide or SAS Management Console. SAS Stored Processes can be accessed via SAS Web Report Studio, SAS Add-In for Microsoft Office, the Web (through the SAS Information Delivery Portal, Stored Process Server Web Application, or Web Services) or by using custom programming. The types of BI content that are produced are SAS Stored Processes code and related metadata.
SAS Web OLAP Viewer for Java – is designed for the power user who has some knowledge of an organization’s OLAP data sources and is comfortable solving business problems by exploring these data sources. The types of BI content that are produced are Data Explorations, Bookmarks, and SAS Reports.

SAS Add-in for Microsoft Office – enables users of Microsoft Office to leverage the power of SAS to access data sources and run SAS Stored Processes from within Word, Excel, or PowerPoint. This functionality is integrated seamlessly and accessed in the same way as any other Microsoft Office functionality (menu options and dialogs). The types of BI content that are produced are Word documents, Excel spreadsheets, and PowerPoint presentations. (Because most organizations have existing mechanisms for controlling or administering these types of documents, discussion about them in this paper is minimal.)

The following table summarizes the interfaces and the BI content that each produces.

<table>
<thead>
<tr>
<th>Interface</th>
<th>BI Content Produced</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS Information Map Studio</td>
<td>Information Maps</td>
</tr>
<tr>
<td>SAS Web Report Studio</td>
<td>SAS Reports and Report Templates</td>
</tr>
<tr>
<td>SAS Enterprise Guide and Base SAS</td>
<td>SAS Stored Processes and SAS Reports</td>
</tr>
<tr>
<td>SAS Web OLAP Viewer for Java</td>
<td>SAS Reports, Data Explorations, and Bookmarks</td>
</tr>
<tr>
<td>SAS Add-In for Microsoft Office</td>
<td>Word Documents, Excel Spreadsheets, and PowerPoint presentations</td>
</tr>
</tbody>
</table>

WHAT IS MEANT BY MANAGING CONTENT?
This paper focuses on the securing, scheduling, automating, archiving, distributing, and promoting (moving content between repositories) BI content. Although many people might be involved in the creation of BI content, usually, most of the activities discussed here are handled by a smaller group of people whose responsibilities include the administration of the reporting environment.

One of the strengths of SAS BI Server is the great flexibility in how it is deployed. In addition to providing a wide range of interfaces, SAS BI Server does not force organizations to work in a specific, limited way. Exactly which management activities any given organization will need will depend a great deal on how SAS BI Server is deployed and the organizational culture that is in place. Some organizations might focus primarily on the ad-hoc capabilities; others will want to integrate BI content creation into established, more defined processes. Most organizations will need to find the right blend of these approaches. Fortunately, SAS BI Server can support all these approaches.

DISTRIBUTION
At the very basic level, distribution can be managed by saving BI content in either a shared location or a private location. Shared locations are accessible to multiple users. Private locations are only accessible by a specific individual. However, SAS provides a number of additional methods to distribute BI content across an enterprise.

SAS WEB REPORT STUDIO
SAS Web Report Studio provides an intuitive, easy-to-use, interactive reporting environment. Report authors can quickly generate new reports or modify existing reports. This ad-hoc reporting capability makes it a powerful tool for helping users solve their business problems. However, at some point, most users will find they need to share their report with someone else. SAS BI Server provides a number of alternatives for meeting this distribution need. The simplest solution is to save the report to a commonly accessible folder in the SAS Web Report Studio repository, and let other people know where it is. They can then log on to SAS Web Report Studio and view the report. If you need to share your report with someone who does not have access to SAS Web Report Studio, what then?

Web-based Access
One option is to use the SAS Information Delivery Portal (part of SAS Enterprise BI Server). Although the SAS Information Delivery Portal doesn’t distribute content per se, it does enable users to access reports that are in the content repository by using SAS Web Report Viewer. This viewer presents the report in a format that is very similar to the format in SAS Web Report Studio. The SAS Web Report Viewer enables the report consumer to take advantage of the interface’s interactivity with the reports such as drilling through data, applying filters, and sorting, but it prevents them from being able to modify the report or create a new report. Users of the SAS Information Delivery Portal can search to find reports of interest, or they can monitor certain folders in the Report repository. In either case, reports can be easily distributed by saving them to repository folders that are accessible to the intended audience.
Publication Channels

If a report author wants to limit the report consumers to a more static view of the report or make it available to people who don’t have access to either SAS Web Report Studio or SAS Web Report Viewer, they can use the Publication Channel functionality. A Publication Channel can be thought of as an electronic distribution list that’s focused on a specific topic, organizational unit, or any other common need. Publication channels can be defined so that membership is centrally controlled or is open so that people can add or remove their own names. An organization can define as many Publication Channels as needed. After the BI content is published to a channel, the content can be accessed by Channel Subscribers via the SAS Information Delivery Portal and e-mail. Publishing to Publication Channels is available interactively from SAS Web Report Studio and SAS Enterprise Guide and, programmatically, from SAS code. In SAS Web Report Studio, the report author can publish content, that is, a report (in the form of a PDF file) by using the SAVE AS dialog with SAS Web Report Studio. The PDF file is put into a SAS Package (a compressed ZIP-like format) and is either attached to an e-mail that’s sent to channel subscribers or added to a library of content on the SAS Information Delivery Portal. To view the report, users open the package (using the SAS Package Viewer if they are not using the SAS Information Delivery Portal) and click the link to the PDF file. In the SAS Information Delivery Portal, users can locate the report package by searching or have it appear on their customized portal pages. In either case, the content that’s accessed is a static PDF file.

Publication Channels provide a powerful tool because they enable the report author to focus on report creation and not worry about maintaining lists of report recipients and their e-mail addresses. Names can be added or dropped from report distribution as part of the standard, centralized, user management process. Also, report consumers can search for content of interest to them, which eliminates the need for report authors to keep track of everyone who might need to see the report.

An administrator creates Publication Channels using the Publishing Framework plug-in to SAS Management Console. In addition to defining the name of the channel and, possibly, who has access to the channel, the administrator can define key words to facilitate searches and identify if and where channel content is archived. The administrator also sets up the required security for the channel.

“Bursting” Reports

SAS Web Report Studio provides another technique for report authors to distribute a static “snapshot” of a report (Figure 1). This technique enables a report to be “burst,” that is, to distribute different portions of the report to different audiences. For example, a sales report could be “burst” so that each sales manager gets a copy of the report that is specific to each manager’s territory and contains nothing about other territories. This is a powerful feature that lets the report author create a single report, run it, and have “personalized” versions distributed to the members of a large audience. Reports that are distributed through this mechanism can be delivered via e-mail or via Publication Channels.

To enable this form of report distribution, administrators create recipient lists that are SAS data sets stored in a specific library and have a defined structure. The data sets can be defined in SAS Web Report Studio or created elsewhere and can be maintained programmatically. These lists define the “bursting” criteria, the recipients of each section, and the mechanism for delivering the content (either e-mail or publication channel).
Using the preceding example of a sales report that’s “burst” based on sales territories, the recipient-list data set might look the following:

<table>
<thead>
<tr>
<th>TERRITORY</th>
<th>EMAIL</th>
<th>CHANNEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>EAST</td>
<td><a href="mailto:Tom@mycompany.com">Tom@mycompany.com</a></td>
<td></td>
</tr>
<tr>
<td>CENTRAL</td>
<td><a href="mailto:Bill@mycompany.com">Bill@mycompany.com</a>, <a href="mailto:Kim@mycompany.com">Kim@mycompany.com</a></td>
<td></td>
</tr>
<tr>
<td>WEST</td>
<td>WesternSalesManagers</td>
<td></td>
</tr>
</tbody>
</table>

In the preceding example, after the sales report is generated (and defined with a group break on the TERRITORY field), the section of the report that deals with the EAST would be sent via e-mail to Tom, and Kim and Bill would receive the section of the report that’s related to the CENTRAL territory. The section that contains sales information in the WEST territory would be published to the WesternSalesManagers publication channel.

**SAS WEB OLAP VIEWER FOR JAVA**

Although the SAS Web OLAP Viewer for Java is designed as an interactive, ad-hoc, OLAP data exploration tool, a user might still have the need to share with other users what was found. If the other users have access to SAS Web OLAP Viewer for Java, distributing content requires nothing more than creating Data Explorations and Bookmarks in a shared location. If the other users do not have access to SAS Web OLAP Viewer for Java, the data exploration can be saved as a SAS Report (which has the same format used by SAS Web Report Studio). After a report is created and saved, it can be accessed just like any other report that’s generated through SAS Web Report Studio. It can also be distributed through any of the distribution techniques that were described earlier.

**SAS INFORMATION MAPS AND SAS STORED PROCESSES**

SAS Information Maps and SAS Stored Processes are different from the other types of BI content because they generate content rather than their being consumable “as is” by an end-user. Sending someone without access to SAS Web Report Studio an Information Map would be pointless—an information map only describes some set of data. Sending a report consumer a SAS Stored Process would be similarly pointless—a stored process is nothing but SAS code and associated metadata. Having said that, administrators might find it necessary to share...
information maps and stored processes with other administrators or make them available to various user communities. This is accomplished by using the SAS BI Manager plug-in to SAS Management Console, which is described in the “Promotion” section later in this paper.

AUTOMATION
Although SAS Web Report Studio is generally thought of as an ad-hoc reporting tool, it does provide features that enable an organization to automate some of its report generation capabilities. After a report is defined interactively, the report can be generated in batch, automatically, without human interaction. A major benefit is that a batch-generated report can be displayed to report consumers more quickly because all the data processing and report formatting has already occurred. In addition to improving response time for end-users, batch generating the report might reduce the demands on computing resources because the report only needs to be generated once instead of every time the report is viewed.

When the SAS BI Server is configured with a scheduling server, it is also possible to schedule when a batch report is generated. For example, when a sales report is defined interactively, it can be scheduled to run each night. When report consumers come in each morning, they can access the latest report. Because the report was batch-generated overnight, the impact of a large number of users bringing up the same report in the morning is minimized. Batch reports can be scheduled directly from within SAS Web Report Studio (Figure 2) or from within SAS Management Console. In either case, the integrated scheduling environment permits a wide range of scheduling options, including:
- running at a specific time;
- running on a specific day of the week or month;
- running every x number of days at a certain time;
- running the third Tuesday of every other month; and so on. The scheduling wizard also enables the user to specify values for any prompts that are associated with a report and to specify the Publication Channel to use if the report will be distributed via that mechanism.

In addition to working with SAS Web Report Studio Reports, this scheduling capability can be used to schedule SAS Stored Processes. SAS Stored Processes are most commonly used to generate reports that require data processing, formatting, or layout options that are not usually handled by SAS Web Report Studio. For example, an organization might have an existing standard report that pre-dates the implementation of SAS Web Report Studio and needs to be carried forward. Another example is a report that might need to include specialized analytical output that cannot be appropriately rendered using the standard SAS Web Report Studio objects. In any case, scheduling stored processes is handled by using the same wizard interfaces.

Even if a scheduling server hasn’t been configured, the ability to batch generate reports can be very useful. It also enables organizations to integrate report generation into existing operational processes. Many large organizations have developed extensive and complex streams of regularly scheduled data processing. By creating batch-generation scripts, the reports that are created by using SAS Web Report Studio can be added to existing processing streams. This capability also enables organizations to re-generate reports based on events (for example, when sales territories are re-aligned or other specific activities take place, generate this report). Generating reports doesn’t have to be based only on the clock or the calendar.

PROMOTION
One of the most common content management tasks is “promotion,” which is moving content from one environment to another. For example, promotion might entail moving a report from a developer playpen or development environment to the production environment used by the business-user community. Because a single report that’s created in SAS Web Report Studio is made up of multiple elements, it’s important to make sure that the report is moved in its entirety. Furthermore, because a report is tightly linked to a specific Information Map, it is also necessary to make sure that an identical Information Map exists in the new environment.
SAS makes these tasks easy by providing import/export functionality in the BI Manager plug-in to SAS Management Console. The BI Manager plug-in includes a wizard-like interface that prompts users through selecting the content that they need to move and identifying where the export file should be stored (Figure 3). The content is exported into a SAS Package file (very similar to a standard ZIP file) that users can move, if needed, to a location that’s accessible to the new environment.

From the new environment, users can import the content by using the BI Manager plug-in and SAS Management Console again (Figure 4). In addition to enabling users to select which SAS Package to import, the Import wizard enables users to specify which content should be imported; how existing content will be handled (overwritten or preserved); and how to map the original environment (application server names and stored process repository locations) to the new environment.

BI Manager handles most types of BI content: SAS Information Maps, SAS Stored Processes; SAS Web Report Studio Reports (including banner and conditional highlighting images); and SAS Web OLAP Viewer for Java Data Explorations. In addition to handling the metadata that’s associated with the selected BI content, BI Manager moves all the necessary physical files. For example, for a given SAS Web Report Studio report, BI Manager moves the XML file that defines the report data and layout (stored in the WebDAV repository); the images used for report banners, footers, or conditional highlighting; and any other images that are used as report content.
ARCHIVING

In many reporting environments, there's a need to take a "snapshot" of a report at some point in time. This might be to satisfy a government regulation or some internal audit requirement. SAS Web Report Studio enables a user to save "versions" of the report and to specify how many previous versions should be retained. At any time in the future, that user or any other authorized SAS Web Report Studio user can review any of the archived reports. The archived reports are saved in PDF files in order to maintain the layout, titles, and footnotes in the reports, in addition to the data values.

The Publication Channel functionality that was described in an earlier section can also play a role in the archiving process. If a channel is defined to include an archive location, everything that’s published to the channel remains in the archive until it is explicitly deleted. Even if publication channels are not being used for report distribution, they can be used as an archive mechanism, and a channel can be defined solely for the purpose of archiving. As reports are updated or modified, report authors could “archive” a snapshot by publishing to the channel. Even if the report is generated in batch, the scheduling wizard (and the underlying batch utility) enables the user to specify a publication channel.

Another archiving mechanism is the BI Manager plug-in that’s described in the discussion of promotion. The BI Manager plug-in enables users to specify which BI content will be exported. The exported content can include SAS Reports, Information Maps, and Data Explorations. SAS Package files that are created by the export process can serve as a snapshot of the content at some point in time. If, at some later time, there’s a need to return things to an earlier state, the content can be imported into the original repository (rather than a different repository, which would be the case when promoting content). One very important difference between this archive mechanism and the others is that rather than a static snapshot of a given report's content at a specific point in time, the export file would contain a snapshot of the definition of the report at that point in time. This means that, instead of showing the sales numbers for a given date, it would show the current state of sales numbers in a report format. To re-create the report exactly as it was, it would be necessary to restore the underlying source data to the state it was when the archive was created. The purpose of this type of archiving is not to capture the state of the report contents, but to capture the layout and settings of the report, in case modifications don’t work out. Some organizations might choose to regularly archive portions of the report repository (using BI Manager) to prepare for this possibility.
SECURITY
The most important SAS BI Server administration task is defining an appropriate security scheme. The best security model will ensure that no users can view reports or access data that they don’t have a valid business reason to view or access, but it should be flexible enough to minimize the need for administrative activity. The most successful SAS BI Server implementations consider the security model before delivering content to the entire user community.

While every organization’s specific security needs are different, the flexible security features in SAS BI Server are designed to support most of them. The SAS BI Server security model enables security restrictions to be set at multiple levels: the lowest data level, the information map level, and the report level; and provides a rich set of specific permissions at both the metadata and physical layer level. In addition to this resource-level security, the SAS BI Server includes role-based security that limits the functions that are available to users through the interfaces.

The SAS BI Server security model is composed of four major elements: resources, permissions, users, and roles. Resources are the “what”, that is, the specific BI content objects that need security protections, for example, the Monthly Sales Report, the Marketing Forecast Information Map, or the YearEndClosing Stored Process. Permissions are low-level restrictions (the “how”) that can be performed on a resource, for example, being able to read a SAS Report or create a report in a specific location. Users and groups are the “who”, the people who are granted or denied access to a resource. Roles are another, very specific type of “how” that control how the interfaces appear and what users are allowed to do. For example, SAS Web Report Studio can be configured so that only people who are defined in “Advance Report Users” are allowed to schedule reports; other users do not even see the schedule options in SAS Web Report Studio. Administrators use SAS Management Console to set access controls, to define users, and to assign users to groups or roles.

BEST PRACTICES
Although each organization and implementation will eventually develop their own specific best practices, we can offer some general suggestions as a starting place.

Define a security policy before content is created.
It cannot be stressed enough that the wrong time to develop a security policy is after users have been allowed to create content. An implementation is more likely to be successful if someone has taken the time to think about the type of content that will be created, who will be creating content, who will be consuming content, and what limits need to be in place to prevent the wrong people from viewing the content.

Control access at the group level rather than at the individual level.
In most organizations, there will be some natural clusters of users who have similar access needs. These clusters can be tied to organizational structure (everyone in HR or everyone in Accounting) or to job function (the actuaries, the sales managers, and so on). Using the User Manager plug-in to SAS Management Console, user groups can be defined and individual users assigned to the groups. Granting and denying access to resources to these groups (rather than individual users) reduces the administrative burden. Consider what might be necessary if a new employee joins the HR Department. If access policies are defined at the individual level, an administrator would need to explicitly grant (or deny) the appropriate permissions for every BI content object to the new employee. If permissions are assigned at the group level, the new employee would be given the appropriate permissions simply by being added to the HR Department group. The SAS security model is very flexible. It allows users to be named in multiple groups, and enables some permissions to be granted at both the group and the individual level.

Apply access constraints at the folder level rather than at the specific content level.
Just as there are natural clusters of users, there will be some natural clusters of BI content. Again these clusters can be defined by organizational structure, similarity of topics (sales forecasts, revenue) or anything else. The SAS security model permits content to be grouped into collections called folders. Access permissions are inherited from the folder level to the individual BI content object. Controlling access at the folder level (rather than the object level) also reduces the administrative burden. Consider again the example of a new employee in HR. By adding a new name at the group level, administrators save time. However, if the administrators need to grant or deny access for that group to every item in the repository, they will quickly be “swamped.” By dealing with access permissions at the folder level, the administrator can easily grant or deny access to a large number of objects. Here, too, the security model enables administrators to set limits at both the folder and object levels, giving them the flexibility to handle a wide range of situations.

Deny access to content by default; grant access only by explicit action.
This guideline is a type of insurance policy. It protects content in case you make a mistake. In most organizations, accidentally preventing someone from getting access to something they need is better than letting someone access...
something that they shouldn't be able to access. In other organizations, the culture might be more open. It might be appropriate to permit most people to access most of the content and deny access only to a small sub-set. In this environment, the default can be set to allow access to content and deny access only by explicit action. In either case, taking the time to understand the patterns of content creation and use will make an administrator's life much easier.

CONTENT FOLDER ARCHITECTURE
Once you have decided who should access what content, the next step is to create a content repository that minimizes the administrative burden. By default, SAS Web Report Studio provides the following two general areas in which users can save content: shared folders, which means that the folders are available to other users; or private folders, which means that the folders are available only to specific individuals. A single shared folder that is used by a large number of users might quickly overflow with reports. This will make it difficult for anyone to find the right report, and difficult to apply the appropriate security restrictions. One solution is to create sub-folders in the shared reports area for the groups that you define. This enables you to easily grant access to the folder to the group. To prevent folders from becoming cluttered with superfluous content, users should be encouraged to use their personal folders as much as possible. The BI Manager plug-in to SAS Management Console defines the folder architecture.

PERMISSIONS
The security model uses permissions to control how users can interact with BI content or objects. Permissions give the authority to read or write content or metadata. The specific permissions that are most relevant for this paper are: ReadMetadata and WriteMetadata. The ReadMetadata permission controls whether a user can view an object. The WriteMetadata permission controls whether a user can create, move, or delete an object. For example, the HR Department content folder might be set up to deny ReadMetadata permission to everyone except the members of the HR Department. In this case, members of other departments would not see the HR Department folder when looking at the content repository. If the ReadMetadata permission was not denied to other groups, they could see the folder but would not be able to create new reports or modify the existing reports in that folder unless they had been granted the WriteMetadata permission.

CONCLUSION
The attractiveness of the interface or the ease with which reports can be created is often the major focus of a discussion about Business Intelligence. However, as important as these topics are, a successful BI implementation, that is, a BI implementation that enables users to better understand their situation and react more effectively to solve their business problems, requires more. This paper discussed a number of (possibly more mundane) activities that are as important to achieving success. By effectively managing the reporting environment and the content that’s created in that environment, administrators can ensure that the right people have the right access to the right information at the right time.

NOTE
The features and capabilities described in this paper are available in SAS 9.1.3 Service Pack 4, SAS Business Intelligence Server (including SAS Web Report Studio 3.1, SAS Web OLAP Viewer for Java 3.1 and Release 2.1 of the SAS Add-in for Microsoft Office) and SAS Enterprise Guide 4.0. Some features and capabilities that are discussed might involve licensing additional products.

ACKNOWLEDGMENTS
The authors would like to thank Scott Sweetland and Rick Styll for their assistance in the preparation of this paper.

RECOMMENDED READING
In addition to the on-line help that’s available from within the products (especially SAS Web Report Studio), the following documentation contains detailed information about the topics discussed in this paper:


CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors:

Greg Smith
SAS Institute Inc.
111 Rockville Pike, Suite 1100
Rockville, MD 20850
301-838-7030
Greg.Smith@sas.com

Diane Hatcher
SAS Institute Inc.
SAS Campus Drive
Cary, NC 27513
919-531-0503
Diane.Hatcher@sas.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.