



**BBCRM DATA OUTPUT OPTIONS:
LEVERAGING THE TOOLSET TO MEET THE NEEDS OF
END USERS WITHIN YOUR CURRENT TEAM
STRUCTURE**

Brandon Ferris

Senior Director of Strategic Services and Fundraising Counsel
Zuri Group

CONTENTS

1. Introduction	4
2. Data Output Options	5
2.1. Listbuilder	5
2.1.1. Advantages	5
2.1.2. Challenges to Overcome	5
2.1.3. Considerations	5
2.1.4. When to Use.....	6
2.2. Datalist.....	6
2.2.1. Advantages	6
2.2.2. Challenges to Overcome	6
2.2.3. Considerations	6
2.2.4. When to Use.....	6
2.3. Smart Query	7
2.3.1. Advantages	7
2.3.2. Challenges to Overcome	7
2.3.3. Considerations	7
2.3.4. When to Use.....	7
2.4. Ad-Hoc Query, Custom Ad-Hoc Query Views	7
2.4.1. Advantages	7
2.4.2. Challenges to Overcome	7
2.4.3. Considerations	8
2.4.4. When to Use.....	8
2.5. Ad-Hoc Query Reports	8
2.5.1. Advantages	8
2.5.2. Challenges to Overcome	8
2.5.3. Considerations	8
2.5.4. When to Use.....	8
2.6. Export	9
2.6.1. Advantages	9
2.6.2. Challenges to Overcome	9
2.6.3. When to Use.....	9
2.7. KPI (Key Performance Indicator).....	9

2.7.1.	Advantages	9
2.7.2.	Challenges to Overcome	9
2.7.3.	Considerations	10
2.7.4.	When to Use.....	10
2.8.	OData	10
2.8.1.	Advantages	10
2.8.2.	Challenges to Overcome	10
2.8.3.	Considerations	10
2.8.4.	When to Use.....	10
2.9.	Custom Reports.....	10
2.9.1.	Advantages	10
2.9.2.	Challenges to Overcome	11
2.9.3.	Considerations	11
2.9.4.	When to Use.....	11
2.10.	User-Defined E-mail Alerts	11
2.10.1.	Advantages.....	11
2.10.2.	Considerations.....	11
2.10.3.	When to Use	11
2.11.	Third-Party Solutions	11
2.12.	Shared Definitions	12

1. INTRODUCTION

Getting data out of any fundraising CRM system in various forms (reports, queries, mailing lists, exports, dashboards, etc.) is a universal challenge in the nonprofit sector, regardless of the CRM system used. Pulling data accurately requires deep and specialized knowledge of the tools, data, database structure, institutional business rules, and conventions of fundraising (particularly when dealing with giving data).

The common objection to enabling self-service ad-hoc reporting is that end users can't be trusted to do it correctly – they don't understand the business rules, they don't have the time and patience to learn to use self-service tools correctly, etc. And there are potential negative business consequences – mailings sent to the wrong people, misreported revenue numbers, user frustration, and, ultimately, loss of confidence in the data itself (often unfounded – usually it is the outputs, not the data, that are inaccurate).

But very few nonprofit organizations have the financial resources necessary to employ enough full-time data analysts/business intelligence specialists to meet the reporting and data output needs of their user populations. Usually it's not even close. As a result, nonprofits *have to* follow self-service-oriented models for getting data out of their systems. Either that or force users to endure long wait times for every mailing list and query they need to do their jobs. The latter option is fairly common, especially in the early period after going live with a new fundraising CRM system.

Even in larger, more robust nonprofit business intelligence environments, some end user data output needs cannot be anticipated in advance – that is to say, there will always be a significant number of ad-hoc (or semi-ad-hoc) data requests that can't be fulfilled with the simple push of a button. A certain amount of end-user input – that is, self-service – will always be necessary. The challenge, then, becomes how to enable end users to get the information they need in a timely manner, using a combination of:

1. Fully pre-built options.
2. Formal requests to be fulfilled by specialists (a topic which deserves its own discussion), and,
3. **Carefully designed self-service options** that allow end users – who are not data experts and who have things other than pulling data to think about – to pull and output data quickly, easily, and with confidence.

The remainder of this white paper focuses on #3 – in particular, how to take advantage of the considerable range of output options offered by Blackbaud CRM (BBCRM hereafter) to meet the information output needs of end users without requiring them to become BI specialists. BBCRM offers a robust toolset, and when applied strategically, they can help users get the information they need with more confidence and less frustration.

2. DATA OUTPUT OPTIONS

2.1. Listbuilder

2.1.1. Advantages

- Easy filtering for end users – much easier than ad-hoc query tool; spreadsheet-like experience.
- Much easier for the end user to select the correct information than in ad-hoc query.
- User can choose which columns to display and the order in which they appear.
- User can save multiple sets of filter and column settings as “lists.”
- Can be placed on tabs for easy access, or can be placed on pages.
- When placed on tabs it can be passed a context ID (for example, the fundraiser’s GUID so it could live on the My Fundraiser Page).
- Action buttons can be added so users can take action, such as editing the record in the row.
- Results can be exported to Excel as is.
- Even the most sophisticated Listbuilder, allowing users to filter every column based on lookup lists, search lists, and query views, takes less time to create than a complex custom report. Since most “reports” are lists, Listbuilders provide a much more efficient method, in terms of developer resource time, for meeting users’ needs.
- Can be built upon SQL from queries built in the ad-hoc query tool or written in SQL Server.
- Once you have built a few of these, the time investment is not much more than for a datalist, and Listbuilders have more functionality.
- By placing a set of Listbuilders or links to Listbuilders on a page, you can create an environment similar to the “self-service reporting” environment you may have had in your legacy environment.

2.1.2. Challenges to Overcome

- Unlike a datalist, it cannot be created entirely through the front end of BBCRM; A Query View List Spec must be created in the SDK (although it is a very low effort/time SDK item).
- Lists cannot be saved as selections; however, it is primarily intended as tool for end users, who are less likely to run business processes requiring selections.

2.1.3. Considerations

- When building the underlying query be sure to include the system GUID for the key field/context ID, as well as any GUIDs needed for action buttons (edit, go to, etc.).
- Rename all output columns in the underlying query.
- Be highly **inclusive** when choosing output columns; You can default visibility to “hidden” and end users can choose which columns to hide and show.
- Filters in the underlying query should not be overly restrictive – end users will apply filters in the Listbuilder itself.
- Make all GUIDs hidden in the Listbuilder Query View Spec.
- *To see a good example of a Listbuilder in action, try **Giving list**, which is located in the **Reports** section of the revenue functional area in BBCRM. This tool is available out-of-the box. Collapse the summary section at first to get a full experience of the list functionality. The summary section uses the same parameters as the list section.*

2.1.4. When to Use

- When a commonly needed set of filterable data useful to multiple users can be defined.
- When the list can be located on a page or tab and/or needs context.
- To create a self-service area for end users.

2.2. Datalist

2.2.1. Advantages

- Relatively easy filtering for end users – easier than ad-hoc query tool.
- Does not rely as heavily on the end user to select the correct information as ad-hoc query.
- Can be placed on tabs for easy access, or can be placed on pages.
- When placed on tabs it can be passed a context ID (for example, the fundraiser’s GUID so it could live on the My Fundraiser Page).
- Action buttons can be added so users can perform tasks, such as editing the record in the row.
- Can be created entirely through the front end of BBCRM.
- It is also possible to create custom Datalists using written SQL and the SDK for more power and flexibility.
- Results can be exported to Excel as is.

2.2.2. Challenges to Overcome

- Unlike a Listbuilder, the end user cannot hide/show or re-order columns, and cannot save lists.
- Datalists cannot be saved as selections; however, it is primarily intended as tool for end users, who are less likely to run business processes requiring selections.

2.2.3. Considerations

- When building underlying query be sure to include the system GUID for the key field/context ID, as well as any GUIDs needed for action buttons (edit, go to, etc.).
- Rename all output columns in the underlying query.
- Be **selective** when choosing output columns; End users cannot hide columns they don’t want to see.
- Filters in the underlying ad-hoc query should not be overly restrictive – end users will apply filters in the Datalist itself.
- Make all GUIDs hidden in the Datalist.

2.2.4. When to Use

- When a commonly needed set of filterable data useful to multiple users can be defined.
- When the needed output columns are modest in number and can be clearly defined in advance (since users cannot modify column order and visibility).
- When the Datalist needs to be located on a tab and/or needs context.

2.3. Smart Query

2.3.1. Advantages

- Relatively easy filtering for end users – easier than ad-hoc query tool.
- Does not rely as heavily on the end user to select the correct information as ad-hoc query.
- Can be saved as a selection for use with business processes such as exports, correspondence processes, global changes, etc.
- Can be created entirely through the front end of BBCRM.
- It is also possible to create custom smart queries using written SQL and the SDK for more power and flexibility.
- Results can be exported to Excel, up to 100,000 rows.

2.3.2. Challenges to Overcome

- Works best when designed with only a few user input fields. If the user must supply more than 2-3 input parameters, the experience starts to feel cumbersome (somewhat more like ad-hoc queries).

2.3.3. Considerations

- Rename all output columns in the underlying ad-hoc query.
- Filters in the underlying ad-hoc query should not be overly restrictive – end users will select parameters when they create smart query instances.

2.3.4. When to Use

- When a commonly needed set of filterable data useful to multiple users can be defined.
- When selections may be needed.

2.4. Ad-Hoc Query, Custom Ad-Hoc Query Views

2.4.1. Advantages

- 100% ad-hoc; end users can select any data available in the query views and are not limited by what was pre-defined.
- Can be saved as a selection for use with business processes such as exports, correspondence processes, global changes, etc.
- Can be created entirely through the front end of BBCRM.
- Some query views can query the data warehouse, which improves performance and simplifies query building.
- Results can be exported to Excel, up to 100,000 rows.

2.4.2. Challenges to Overcome

- The ad-hoc query tool is too complicated for many end users, and it is often difficult for them to confidently pull the correct results.

2.4.3. Considerations

- Push users toward **data warehouse query views** (BBCRM provides several out of the box) as much as possible. They are simpler to navigate, provide less possibility for error, perform much faster, and reduce the likelihood of accidentally creating self-referential queries that don't run to completion and affect system performance. Consider restricting access to certain query views through security.
- Simplified, **custom query views** can be built as another way to provide users with simpler experiences. *We **highly recommend** simplifying key out-of-the-box query views into custom query views. In our experience, much end user confusion results from not knowing where to find the fields to query on and related issues.*
- Provide as much formal training as possible.

2.4.4. When to Use

- When you need to pull based on a set of criteria that does not exist in any of the aforementioned pre-built options.
- When selections will be needed.

2.5. Ad-Hoc Query Reports

2.5.1. Advantages

- Can be formatted for sharing with external consumers.
- Uses ad-hoc query built in BBCRM as data set, including ad-hoc queries built using data warehouse query views.
- User can go into Report Builder to group, aggregate, and format.
- Faster to build than full custom SSRS reports.
- Report Manager allows subscriptions; can be used as a modest-effort way to push more sophisticated outputs than e-mail alerts.

2.5.2. Challenges to Overcome

- The combination of the ad-hoc query tool and the Report Builder requires a skill level above that of the average end user.
- Requires some initial setup.

2.5.3. Considerations

- User must download Report Builder 3.0.
- It may be necessary to provide specific permissions.

2.5.4. When to Use

- When you need a simple formatted custom report with simple summarization.
- When you need to push information slightly beyond what can be done with an e-mail alert.

2.6. Export

2.6.1. Advantages

- Provides the finest level of control over output fields through the front end, especial one-to-many fields, which can be exported as separate columns with additional filters (beyond the *record* selection criteria in the underlying query) applied.
- No row limit.
- Can be scheduled.
- Export definitions can be built by power users for use by end users who can reuse them by simply changing the selection being passed.

2.6.2. Challenges to Overcome

- Can be slightly complicated for some end users.

2.6.3. When to Use

- When a spreadsheet is the desired end product.
- When you need to output data including one-to-many fields while still including only one row per record.
- When you need to output very large files above the query tool's row output limit.
- When you need very fine control over output fields, including the ability to further filter one-to-many fields.

2.7. KPI (Key Performance Indicator)

2.7.1. Advantages

- Provides intuitive, user-friendly visualizations of data.
- Relatively easy to create.
- Can be scheduled.
- Can be placed easily on dashboards.
- Several popular metrics, such as average days in plan stage for fundraisers, are available out-of-the-box.
- Custom KPIs can be created in the SDK.

2.7.2. Challenges to Overcome

- Each KPI is designed to do one thing and show a single data point, so they must be narrowly matched with the business objective.
- Some limitations on what can be accomplished using out-of-the-box KPIs.
- While all data outputs are limited by “how the data goes in,” several popular KPIs are particularly reliant upon consistent data entry and business processes aligned with BBCRM's designed functionality. Since the reality in many organizations is often considerably “looser” than that, some KPIs won't have the underlying data to make them work as designed.

2.7.3. Considerations

- KPIs need to have a job schedule set to process them regularly.
- Watch out for accidentally creating a set of KPIs every time you create a marketing effort.
- Determine a place to surface them.

2.7.4. When to Use

- When you want a simple visualization of at-a-glance performance data that can be handled by an out-of-the-box KPI definition.

2.8. OData

2.8.1. Advantages

- Once a connection has been set up, OData allows the user to manipulate real-time data from a dynamic ad-hoc query in Excel (or other tools) through a direct connection to the database. There is no need to go back and re-run an export process to get the most up-to-the minute data.
- Creating an OData connection from an ad-hoc query is very easy to do.
- Once data is in Excel, users can leverage the power of a number of user-friendly Excel plug-ins to create powerful reports even more easily than using Excel alone.

2.8.2. Challenges to Overcome

- Application root URL must be set in Administration.

2.8.3. Considerations

- Since it is based on an ad-hoc query, adequate skill with the ad-hoc query tool is necessary, or the ad-hoc query must be built for the user.

2.8.4. When to Use

- When you need to manipulate and pivot data in Excel using up-to-the-moment data.

2.9. Custom Reports

2.9.1. Advantages

- Final product can be exactly as you want it.
- Able to include complex calculations, rollups, summaries, drill-downs, etc.
- Custom reporting tools may offer the ability to create sophisticated visualizations of data.
- Able to include subreports – “all-in-one” reporting to an extent.
- Can be highly formatted for external consumers.
- Some senior leaders are accustomed to viewing data this way and expect formal reports.

2.9.2. Challenges to Overcome

- Long development time.
- Leadership doesn't always understand why custom reports require so much time from conception to completion; expectations must be well managed.
- A lot of time and effort – not just that of business analysts and developers – goes into specifying, creating, testing, refining, and deploying reports, after which it is quite common for such reports to fall quickly into disuse when the leadership that requested them leaves for other institutions. Within a short period of time, whether anyone even uses certain reports is unknown, along with why and for whom they were originally created.

2.9.3. Considerations

- Requires a formal process for requesting, specifying, developing, testing, and deploying.
- To the extent possible, limit development of highly complex, labor-intensive custom reports to meeting reporting needs that can reasonably be expected to persist through multiple leadership changes.

2.9.4. When to Use

- When you need complex, formatted reports containing calculations, summaries, sophisticated logic, subreports, drill-downs, etc., that cannot be produced in any other way.

2.10. User-Defined E-mail Alerts

2.10.1. Advantages

- Can be created through the front end using user-defined datalists.
- Uses the same front-end job scheduling functionality as other business processes.
- Query output fields become available as e-mail body merge fields.

2.10.2. Considerations

- May send high numbers of e-mails; create query and job schedule accordingly depending on recipient preferences.
- Not contextual/user-specific.
- Like Notifications, it is easy to create too many alerts and inundate recipients; use sparingly and select recipients carefully.

2.10.3. When to Use

- When you need a simple alert sent by e-mail based on additions or changes to data that can be reflected in a query result.

2.11. Third-Party Solutions

A number of third-party solutions exist that are designed to or can be built out to create user-friendly ad-hoc data pulling experiences (similar to Listbuilders or simplified custom query views). Some third

party solutions provide both simplified ad-hoc reporting and powerful custom reporting and visualization capabilities. Examples include Michelangelo (purpose-built end user ad-hoc data tool that includes the creation of a simplified data model), Tableau (comprehensive BI solution primarily used for custom reporting with sophisticated data visualizations; can be used to provide ad-hoc data access with simplified data models), and BizInsight (end user-friendly Excel add-on for easy report creation; can be used with BBCRM OData functionality), to list a few that are known in the nonprofit industry.

2.12. Shared Definitions

This section does not deal with a type of solution, per se, but in order to have a robust reporting environment, it is crucial to have agreed-upon, shared definitions of key terms and concepts, such as revenue vs. recognition, recognition credit types, the many dates associated with gifts, transactions vs. applications (splits), and cash vs. commitments. These should be well documented, easily accessible (in tools such as single sheet job aids), and thoroughly covered in training. In addition, it may be useful to identify and implement ways to tag data, such as adding flags indicating whether a revenue application is considered cash, commitment, or both, in order to “bake in” key definitions within the data (in query views, data warehouse tables, etc.) for easier reporting.