KnowledgeSync™

V10.6.5 User Manual

Contents

1   Configuration

1.1   Installation

Please consult the [KnowledgeSync Version 10 Installation Manual](#) for installation help. The application must be properly installed before beginning configuration.

1.2   Welcome to KnowledgeSync!

Welcome to KnowledgeSync (KS), the industry's leading Business Activity Monitoring (BAM) technology.

This guide helps users configure and use KnowledgeSync according to the needs of their organization. We hope you enjoy working with KnowledgeSync and encourage you to [contact us with your feedback](#). Now let's get started with KnowledgeSync!

1.3   Log into the KnowledgeSync Portal

Using a web browser, log into the KS portal (default user admin; password is password). *Please note:* *Apple's Safari®* *browser is not supported.* The home window opens:
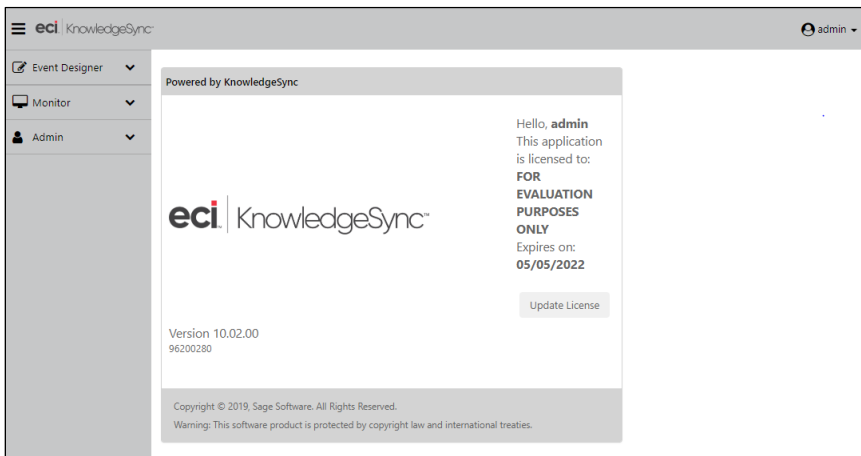
Clicking on the KnowledgeSync name in the top-left of the portal always returns the user to the home window. Below the name are three buttons that open the main KS modules:

- **Event Designer**: KnowledgeSync event and subscriber configuration.
- **Monitor**: Audit trail of KS events that are running and have been run.
- **Admin**: KnowledgeSync configuration and maintenance settings.

1.4   Install EventPak(s)

KnowledgeSync includes pre-configured collections of events (called EventPaks) for over 70 popular front-office and back-office applications. These are available from the same download page as the KnowledgeSync application.

There are often multiple EventPaks for the same application – sometimes for different versions and sometimes for different types of events. Feel free to download and review any EventPaks.

To install an Eventpak:

- Expand the Event Designer option, click on the EventPak option and select **Install.**
- Click on the **Choose File** button and locate the EventPak to install. (First, download the desired EventPak(s) from the KnowledgeSync website.)
- Click on the **Upload EventPak File** button. General information about the EventPak and its origin is shown.
- Click on the **Applications** tab to see a list of the individual KS applications that make up the EventPak, as shown in the EventPak Installation window:
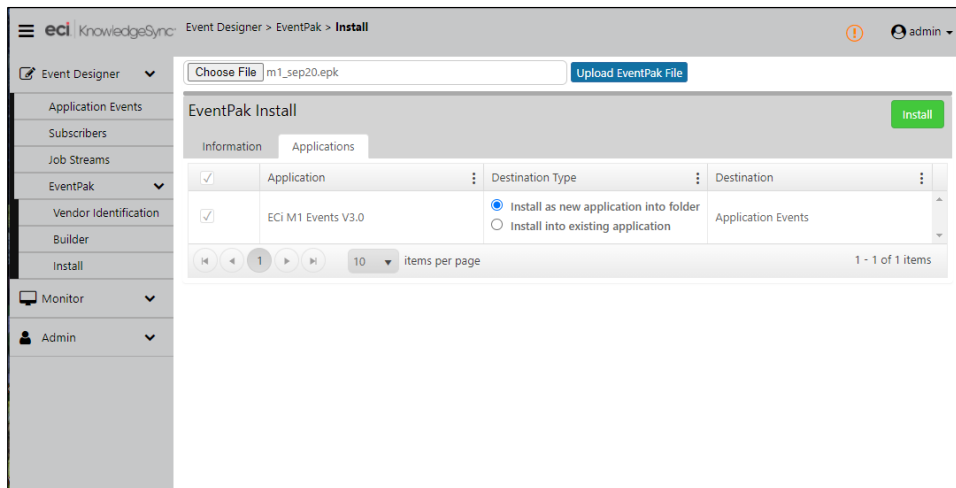
- Click on **Install**. The EventPak will be loaded into KnowledgeSync, and you will receive a confirmation message when this process is complete.

1.5   Using KnowledgeSync to Monitor Application Data

There are four ways that KnowledgeSync can connect to and monitor data within application databases (web-based and on-prem) and within websites:

- .NET Data Provider (preferred & most efficient connectivity method for on-prem application databases)
- ODBC (32bit)
- OLE/DB
- REST API web service (for monitoring hosted apps and data from websites)

The REST API features are discussed in their own chapter, as web services may be used both as the triggering mechanism for a KS event, as well as an action that a KS event executes because of being triggered.

1.6   Initial Set-Up

By now you should have successfully installed KnowledgeSync using the instructions in the Installation guide.

The next steps in the set-up of KnowledgeSync are:

- Connecting KnowledgeSync to the application(s) you wish to monitor.
- Creating & testing the email account that KnowledgeSync will use for the sending of email alert messages.
- Creating a "subscriber" (alert recipient) record for yourself to use during event testing.

The following sections will detail connecting KnowledgeSync to an on-prem database; for details on connecting KS with a web or cloud-based app, please see the chapter of this guide titled "REST API Triggers".

1.6.1   Step #1: Connect KnowledgeSync to Your Application(s)

KS uses events to monitor certain data conditions and respond to those conditions by triggering any combination of alerts, reports, and workflow (data updates). All KnowledgeSync events are grouped by "application".

An application may refer to a specific product (such as "My ERP Solution"), a specific module ("Accts Receivable"), or any source of data being monitored ("Manufacturing Newsfeed").

Expand the **Event Designer** option and then click on **Application Events**. This presents you with a list of all the applications currently defined within KS; this includes applications that have been created through the installation of one or more EventPaks, as shown in the following illustration:
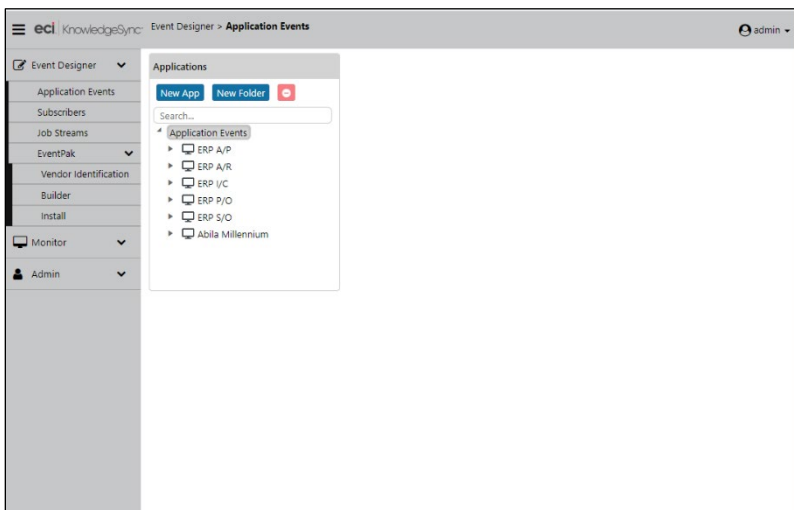
To connect KnowledgeSync to an application:

- Click on the name of the application you wish KS to monitor. An Application Properties window opens:
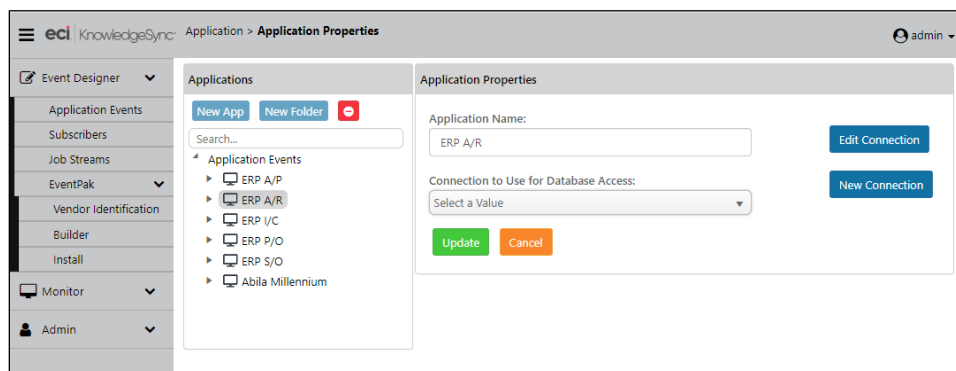
- Click on either **Edit Connection** (if a connection has been specified) or **New Connection** (if no connection has yet been specified) and the applications' Connection Definition window opens:
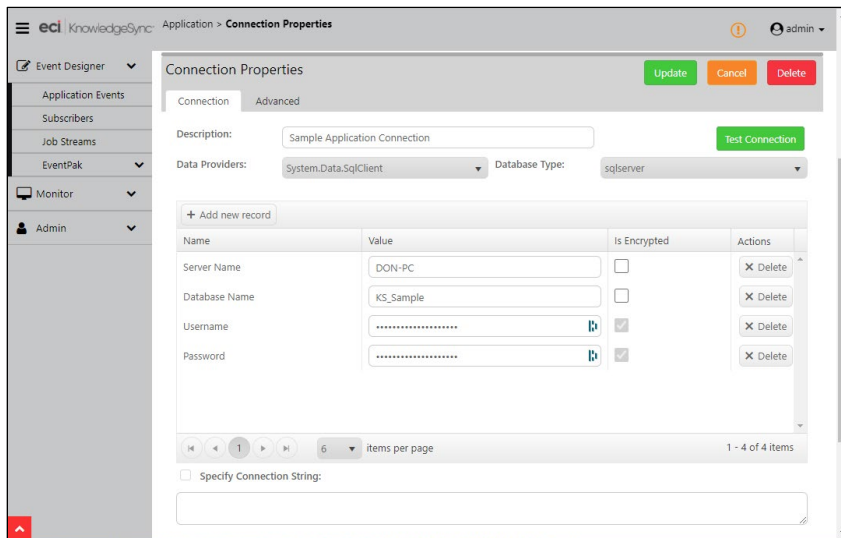
- Select the Data Provider and Database Type and then fill in the appropriate connection information.
- Click the **Test Connection** button. If successful, save the record.
- The options in the **Advanced** tab should be used only on recommendation of the KnowledgeSync support staff; these options are typically used when KS needs to accommodate special database tables, columns, or identifiers.

  Any visual basic script can retrieve the script API values that are configured in the advanced sub-tab, using the method KSGetEvent in a VB script.

  Calling KSGetEvent (apiname) from a script retrieves the name of the API as configured in this option associated with the application that contains the script; the other values retrieve the API user name and password.

Note the following about .NET Database Provider, ODBC and OLE/DB connections:

- Database Provider (preferred & most efficient connection method):
  - All currently installed database providers appear from the drop-down list.
- ODBC:
  - Only system DSNs (sources) may be used.
  - Do not reference mapped drives in any sources (use UNC format instead)
  - Leave the date separation, long table name, link/join prefix and suffix, and column alias fields with their default values.
- OLE/DB:
  - All currently installed OLE/DB sources appear from the drop-down list.

1.6.1.1   File Encryption Configuration

To ensure security, customers are encouraged to encrypt their KnowledgeSync connection strings. (The default is unencrypted.) Below are instructions for both Portal and Desktop Service customers on how to encrypt connection strings.

Portal:

1. Right-click on the KS *Portal* icon on the desktop.

2. Select **Open File Location** and click on the **KSPortal** folder.
3. Right-click on the *Web.config* **file and** open with Notepad (Run as Administrator) or Notepad++.
4. Change the flag **<add key="EncryptConnectionString" value="false" />** to **<add key="EncryptConnectionString" value="true" />** and save.
5. Close the browser and open the portal in browser to login again. The change takes effect after logging in again.

Desktop Service:

1. Right-click on the *Desktop Service* icon on the desktop.
2. Select **Open File Location** and click on **KSService** folder.
3. Right-click on *KSService.exe.config* file and open with Notepad (Run as Administrator) or Notepad++.
4. Change the flag **<add key="EncryptConnectionString" value="false" />** to **<add key="EncryptConnectionString" value="true" />** and save. The change takes effect when restarting the service.

1.6.1.2   If You Need to Create a New Application

To use KnowledgeSync with an application for which there is not a pre-defined EventPak, create an application by following these steps:

- Under **Event Designer**, click on **Application Events** and then on the **New App** button. The following window opens:
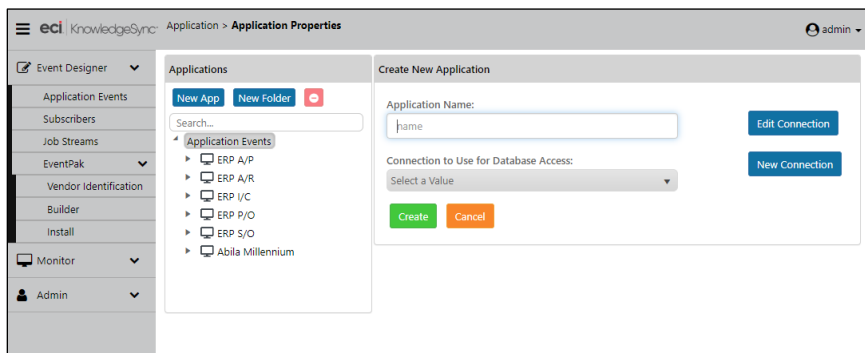
Notes:

Users can add folders as well as apps – folders allow users to group similar applications together. For example, there can be an ERP Solution folder, containing multiple applications, all of which are a part of your ERP application.

The sections in the Application Events module can be resized to allow for maximum available screen space when designing or editing. Use the splitter to resize the layouts for Queries, Events, Reports and Web APIs.

- Specify the name of the created application (e.g., HR Application). Then, click on the **New Connection** button to specify the means of connecting KS to this underlying data – .NET Database Provider, ODBC or OLE/DB and enter the corresponding credentials.

- Create the new application and view it listed under **Application Events**.

1.6.1.3   Learn KnowledgeSync Using the Sample Application

To learn how KS works, KnowledgeSync provides a small, sample ERP database, eventpak and instructions in a Tutorial and Certification kit which is downloadable from:

https://www.knowledgesync.com/downloads/downloadable-assets

1.6.1.4   Creating Multiple Branches for the Same Application

If you're monitoring multiple "parts" of an application – such as an ERP application, in which you have accounts receivable, accounts payable, and the like, you may find it easiest to organize your events into multiple applications – a separate branch for each application "module" you wish to monitor.

Do not create a new "connection" for each application module; instead create a single connection record and use it in all the branches that all point to the same database.

When adding successive application branches in KnowledgeSync, the "connection" window contains a field called "Connection to Use for Database Access" and there is a list button next to it. Click on that button and select the same connection that was used previously for the database you wish to monitor.

1.6.1.5   Test Your Connection

When you are in the "Connection Properties" window, the "Test Connection" button will confirm whether your input connection details (for .NET Database Provider, ODBC or OLE/DB access) are successfully configured.

1.6.2   Step #2: Set Up an Email "Sending" Account

Please see section 2.7.1 for "quick configuration" options for the email sending account.

KnowledgeSync requires you to configure at least one email account from which it sends alert notifications.

Although most organizations configure only a single email sending account, the ability to define multiple sending accounts allows KS to send out messages from different origins (e.g., some alerts from the sales department, others from the service department) depending on the type of alert being sent.

We recommend using the "default configuration" buttons at the bottom of the email account configuration window.

To configure an email sending account, follow these steps:

- From the KS main window, expand the "Admin" branch at the far left, expand the "Services" sub-branch, and then select "Email Accounts." The Email Accounts window opens:
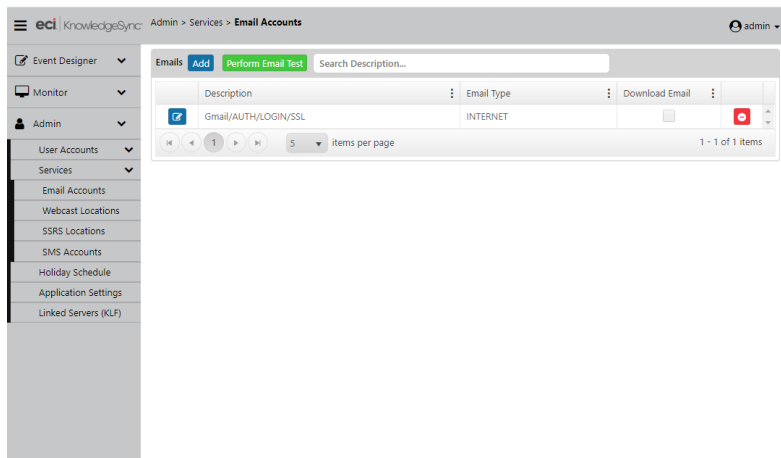
- If there are no sending accounts configured, click on the "Add" button; if you wish to review the configuration of an existing email sending account, click on the blue "Edit" icon to the left of the record you wish to edit.
- The Email Sending Account window opens:

EMAIL SENDING ACCOUNT -- CONFIGURATION

1.6.2.1   Email Account Configuration Buttons

There are five buttons at the bottom of this window to help you with the configuration of your email sending accounts in KS. These options auto-fill most of the settings needed to work with various email systems, and all that is needed is the login (username), password, and a few additional details (such as the SMTP send from name).

Click on the button that corresponds to (or is most similar to) the email system you wish KS to use and fill in the details that are required.

1.6.2.2   Email Account Settings

Here is a brief explanation of the email sending account components:

- **Email Type:** Choices are Internet, Exchange, MAPI, VIM, and Unknown. We suggest using the type of "Internet" for easiest configuration and most flexible usage.
- **Account Description:** A description of this account, this field also specifies the authentication type which is auto filled by the buttons at the bottom of this window.
- **Login Name:** The login username / email address.
- **Password:** The user's email address password.
- **Active:** Whether this account is available for sending email.
- **TLS**: Transport Layer Security. Please see your email administrator for details. Auto filled by the buttons at the bottom of the window.
- **Authentication Methods**: Please see your email administrator for details. Auto filled by the buttons at the bottom of the window.
- **Outgoing SMTP Server Name:** The name of the server that will send e-mail.
- **SMTP From Name:** The name that appears as the "sender" of the e-mail. This is required for type INTERNET and it must be a valid address with a valid domain name. E.g., johndoe@knowledgesync.com.
- **SMTP Port:** Leave this field with its default value unless otherwise instructed.
- **Incoming mail (IMAP) Server Name:** The POP Server if monitoring inbound email (see Email Response System).
- **IMAP Port:** Leave this field with its default value unless otherwise instructed.

1.6.2.3   Email Account Notes

- We recommend using type "Internet" with Exchange if the exchange server is configured for SMTP and POP3.
- HTML-formatted email alerts are supported only with an email type of Internet.

- You may find that you can send email within your domain, but not outside of it. This is due to your SMTP Server not having "Relay' turned on. Once you set the server to allow "Relay", verify that no servers other than the servers within your domain can use that SMTP server for relay purposes.

1.6.2.3.1   Authentication Types Supported

Authentication is enabled for Internet-based email systems. You have six options:

- SSL: choose Port 465
- LOGIN:
- NTLM: choose Port 25
- MSN:
- CRAM-MD5:
- AUTO:

Please refer to your email administrator regarding the type of authentication required by your email system.

1.6.2.3.2   Test Your Email Sending Account

Once you have created an email sending account, follow these steps to test it:

- Make sure the KnowledgeSync Windows service is running.
- From the Email Accounts window, click on the account you wish to test.
- Click on the "Perform Email Test" button. Key in your own mail address as the recipient of this test message and click on "OK".
- Check the KnowledgeSync Monitor branch called "Email Delivery." You should see your test email go from "Pending" to "Sent Today". If the e-mail appears in "Errors Today" (or never leaves "Pending"), there is an error in the configuration.
- If the email fails, stop the KnowledgeSync Windows service and review the email account set-up. Once you have re-configured the email account, you do not need to re-submit the email test as the previous test is still "pending." Simply re-start the service and see if the pending message gets sent.

1.6.3   Step #3: Define Yourself as an Alert Subscriber

The next step is to define yourself as a subscriber – a person who is eligible to receive alert messages. To do so, expand the **Event Designer** option and click on the **Subscribers** sub-branch. The following window opens:
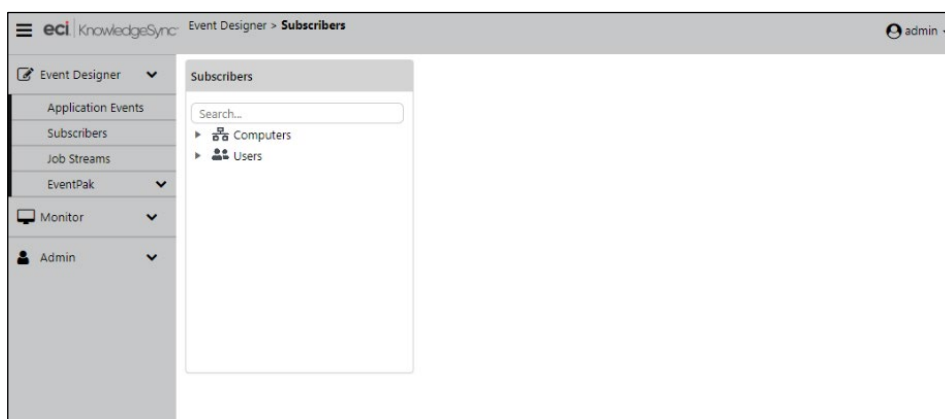
Expand the "Users" branch and you'll see subscriber groups, as in the following window.
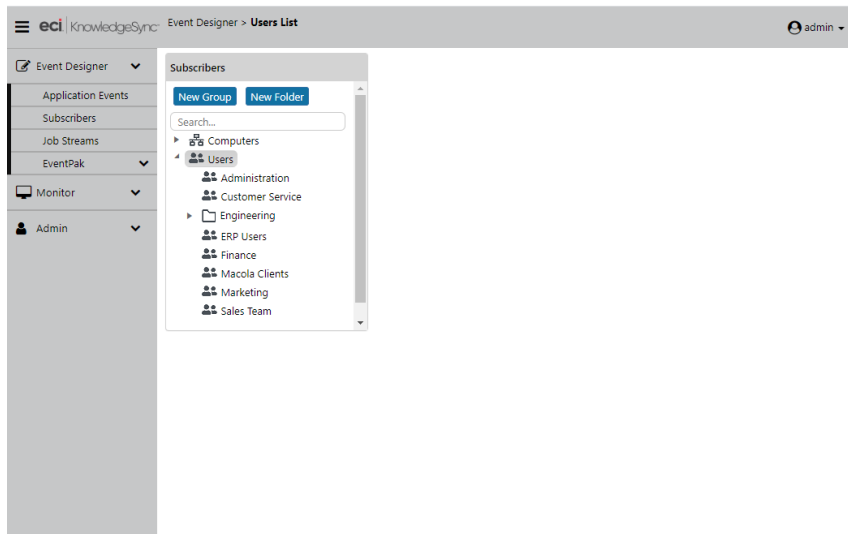
SUBSCRIBER GROUPS

There are some default groups that come with KnowledgeSync; you can use them, change their names, delete them, and add in groups of your own. (More details on subscribers are located in the "Subscribers" chapter of this manual.)

For now, let's add yourself as a subscriber to one of the groups and set yourself up to be able to receive email alerts. Single-click on your selected group and you'll see a window like the following:
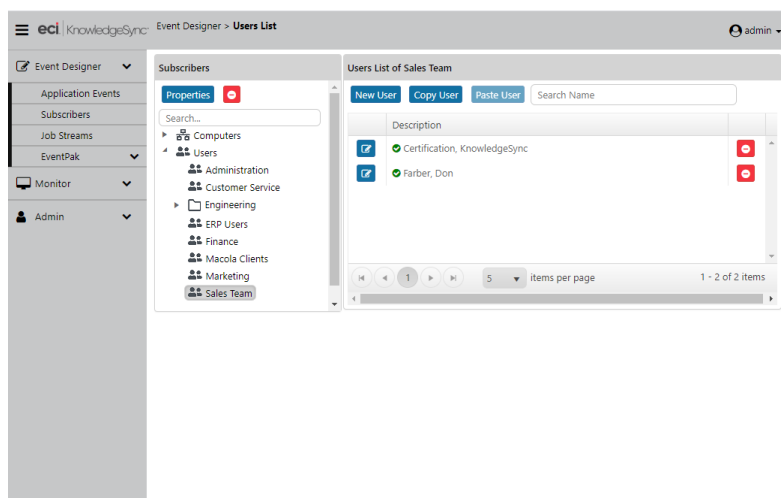
USERS IN A GROUP

Click on "New User" and you'll see the following window:

NEW USER (SUBSCRIBER): GENERAL INFO

Fill in the five fields on the left of the "Description" tab, make sure the "Active" box is checked, and click on the "Email" tab. Enter your most-often used email address into the "Primary" address field, click on the weekdays, and leave the daily time range with their defaults to indicate 24 hours a day. Optionally, fill in either the same email address or a different address in the "Secondary" part of this window. Click on "Save".

USER EMAIL ADDRESS DEFINITION

2   Administration

2.1   KnowledgeSync Administration

The KnowledgeSync Administration Module lets you:

Specify who has access to KnowledgeSync and what functions they can perform (this includes the designation of multiple "tenants" in a multi-tenant environment)
Configure settings for sending alerts via email, web dashboard (webcast) and text (SMS)
Designate the location(s) for reporting via SQL Server Reporting Services (SSRS)
- Specify Integration details for connecting KS to Synergy (If you are using the Exact 'Synergy' product)
Specify your organization's holidays (which let you control whether individual events will run on holidays)
Set application settings for such items as inactivity time and the location of log files.
Create one or more SQL Server "linked servers" – used to connect to database types that have some limitations in terms of data analysis and retrieval

2.2   The Administrator: Logging In

Once you have logged into KnowledgeSync, expand the "Admin" branch and you'll see a window like the following:
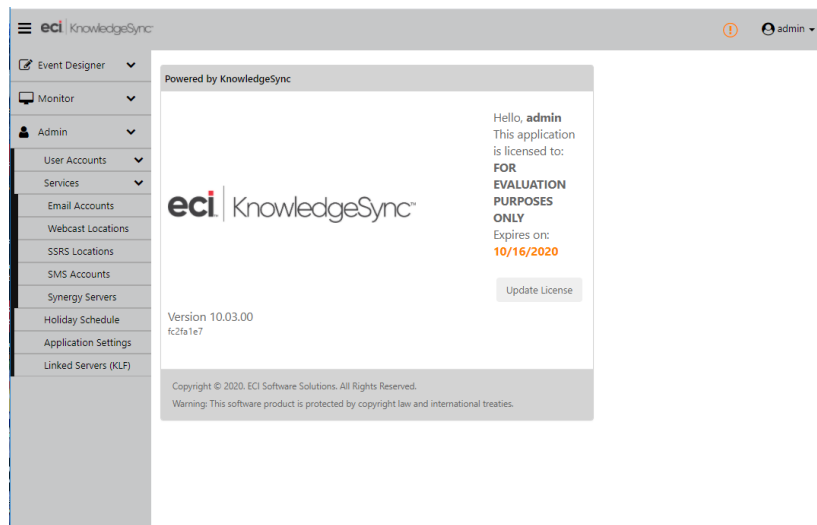
KNOWLEDGESYNC ADMIN OPTIONS (ON LEFT)

2.3   User Accounts: Roles & Users

Expand the "User accounts" option and you'll see options for "Roles" and "Users". It's important to understand the difference between these in KnowledgeSync:

Roles: This controls what KnowledgeSync functions a person can access – such as creating events, viewing events, creating users, et cetera.
Users: This controls what KnowledgeSync data a person can access – for example, user 'A' might have access to KnowledgeSync events for application #1, whereas user 'B' might have access to KnowledgeSync events for application #2. Each KnowledgeSync user is assigned a specific role – and that role controls what KS options and functions that user can access within that person's allowed data sets.
You can copy and paste both role and user records using the "Copy" and "Paste" buttons that appear at the top of each option.

2.3.1 Multi-Tenancy Access Rights

In a multi-tenancy environment, where a single KnowledgeSync installation is being used by multiple customers, there will typically be a single overall KnowledgeSync System Administrator who can grant access rights for all tenants. This "KS Admin" will typically create individual customer-specific KS administrators, such as "KS Tenant #1 Administrator", "KS Tenant #2 Administrator" and so on. The KS System Administrator is responsible for specifying what data sets each tenant administrator has access to, as well as what system options each tenant administrator can access.

It then becomes the responsibility of the individual KS Tenant Administrators to create the user profiles for their respective users – controlling what KnowledgeSync functions each of their users can access.

2.4 Roles

(KnowledgeSync comes with a "Global Admin" role which is a super-user role that has all entitlements enabled. If you choose to set up multiple access levels in your organization, please create new roles and users. You cannot modify or delete the Global Admin role.)

KnowledgeSync "roles" control what application functions a KnowledgeSync user can perform. For each KnowledgeSync "user", you're able to assign them a specific "role" which controls the KS system functions that they can access.

KS comes with a couple pre-configured roles, and you may use (& customize) them as well as create an unlimited number of additional roles of your own. To create a role, follow these steps:

Under "Admin", expand "User Accounts" and then click on "Roles". You'll see a window like the following:
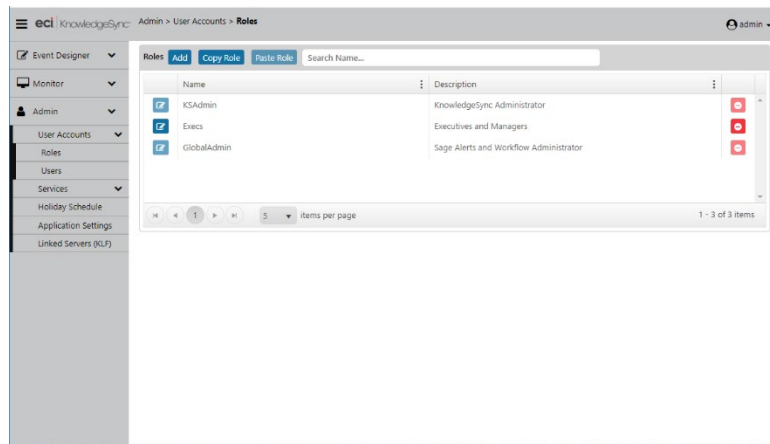
KNOWLEDGESYNC "ROLES"

Click on the **Add** button to add a role and the following window opens:
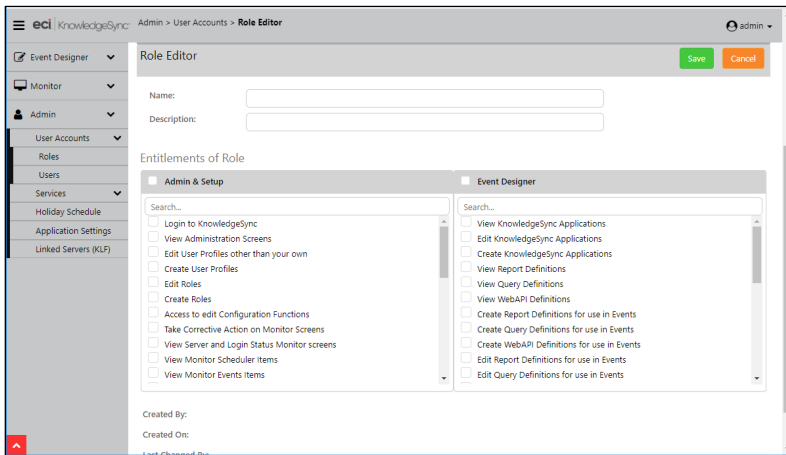
Roles consist of the KnowledgeSync administrative and event design functions that can be performed. Administrative actions are divided into the following abilities:

- Log into KnowledgeSync
- View KS Admin options
- Create and edit both roles and users
- Access KS configuration settings
- Access to various parts of the KS Monitor (including "corrective actions" on any KS events that have errored (such as fixing an invalid email address for an alert)
- Update KS version 10-specific settings

Event design functions are divided into the following abilities:

- View, edit & create KnowledgeSync applications, queries, events, reports, schedules, subscribers, and web APIs
- Build and install EventPaks

- Put a checkmark next to each Admin or Event Design option that you would like to enable a specific role to be able to perform or put a checkmark next to the headings of **Admin & Setup** or **Event Designer** if you would like to enable a role to perform all of the options below that heading.

- Click on "Save" to retain your entries

2.5   Users

(KnowledgeSync comes with a default Admin user which you cannot modify or delete except for the Admin password.)

The Users option lets you specify what data elements (applications, events, queries, et cetera) each person who logs into KnowledgeSync will be able to access. To create a user, follow these steps:

Under Admin, expand **User Accounts** and then click on **Users**. The following window opens:
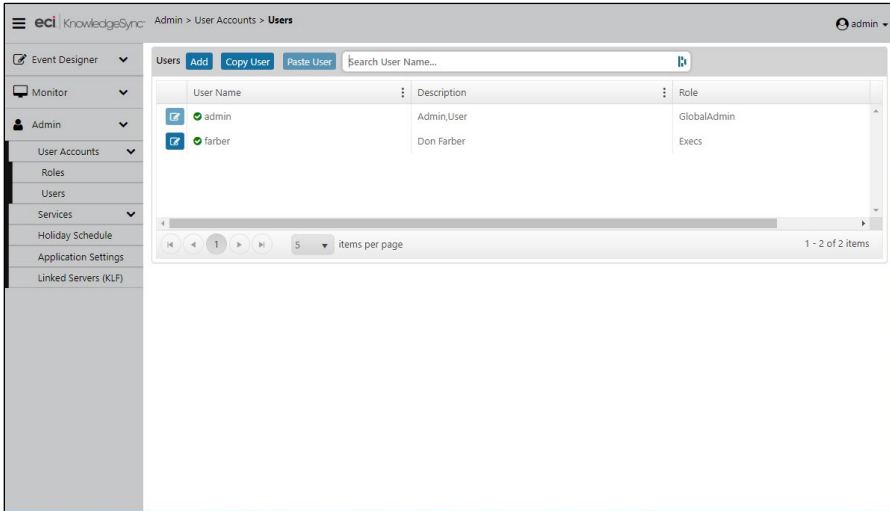
Click on the **Add** button to add a new user and the following window opens:
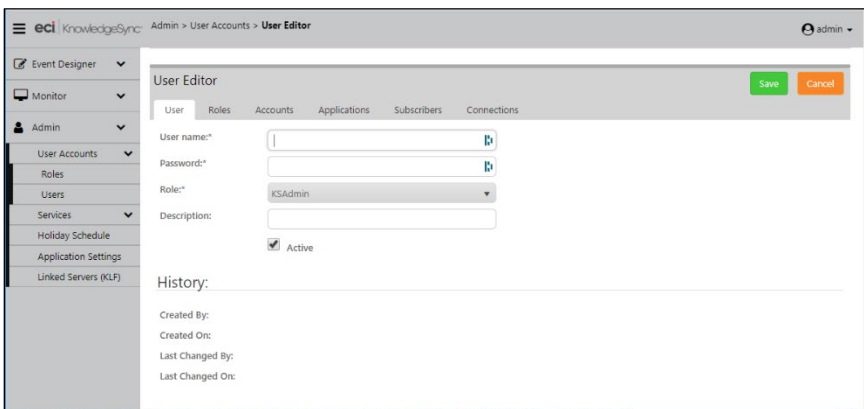
In addition to a username and password, each user is assigned to a specific "role", as detailed in the preceding section. Begin on the "User" tab to specify the person's login name, password, selected role, and description – such as their full name. Note that you can control whether a user is "active" – and thus can access the selected data elements.

2.5.1   Users: Roles
The "Roles" tab within the "Users" option controls the roles that this user can assign to others as well as the roles that this user is able to administer.

2.5.2   Users: Account Access
The "Accounts" tab controls which accounts (if any) a user can utilize in the configuration of KS events that use email, webcast, and SMS for the sending of alert messages, and that use SSRS for the generation of reports. This includes from which:

- email account(s) the user can select
- webcast locations the user can select
- SSRS reporting servers the user can select
- SMS account(s) the user can select

KnowledgeSync uses Twilio as its SMS Provider and users will need to purchase a Twilio subscription to send text-based alerts. Just two fields from Twilio (account SID and auth key) will need to be entered into KnowledgeSync

If you wish to allow a user to select from all email, webcast, texting locations, and SSRS servers, click in the checkbox next to "All Current and Future Accounts".

2.5.3   Users: Application Access

This tab shows a list of all the applications (e.g., ERP, CRM, HR) that are currently configured to be monitored by KnowledgeSync. If you wish to allow a user access to all applications, click in the checkbox next to "All Applications". Otherwise, place a checkmark next to only those applications whose data and events you wish the current user to be able to access.

This is particularly important in multi-tenant implementations, as this allows you to ensure that each KnowledgeSync tenant has access to only those events created for their organization.

Control over which components within an application a user may access or modify depends on that user's "role".

2.5.4   Users: Subscribers

This tab shows a list of all the alert recipients ("subscribers") that are currently available to be maintained as well as selected (within an event) to receive alerts from KnowledgeSync. Note the following:

Subscribers may be individual people, groups of people, and "computers" – locations to which reports and files may be FTP'd.
The subscribers referred to here are "standard" subscribers, i.e., people or groups who may be selected from a list. An event may also deliver alerts to "advanced" subscribers; these are people who are associated with the triggered condition (such as "the customer" or "the salesperson assigned to that customer".
If you wish to allow a user to configure as well as select (within events) all subscribers, click in the checkbox next to "All Subscribers". Otherwise, place a checkmark next to only those subscribers whose data and events you wish the current user to be able to access

2.5.5   Users: Connections

This tab shows a list of all the "connection" records (.NET data providers and ODBC and OLE/DB data sources currently configured within KnowledgeSync). If you wish to allow a user access to all the connection records, click in the checkbox next to "All Connections". Otherwise, place a checkmark next to only those connections whose configurations you wish the current user to be able to access and modify.

2.6   Services

The Services option lets you specify the following:

- The **email accounts** from which KnowledgeSync will send alerts.
- The **webcast** (web dashboard) locations to which the dashboards will be published (as well as the templates for the dashboards' appearance).
- The location of your **SQL Server Reporting Services** (SSRS) (if you will be using KnowledgeSync to generate and deliver SSRS reports).
- The **SMS text** messaging provider, account and authentication details for sending text message alerts.
- The location of your **Exact Synergy** application (if you are integrating KS with Synergy).
- The **holidays** for your organization (so you can control whether KnowledgeSync events will be run on a holiday).
- General KS **application settings**, such as for diagnostic logs and file backups.
- Whether you are using **Linked Servers**, which employs SQL Linked Servers to access data from database with limitations on data analysis and retrieval.

2.7   Email Accounts

The option for Email Accounts lets you define one or more accounts from which KnowledgeSync will send email alerts. Multiple email accounts allow you to tailor the "send from" address, such as from different companies or different departments within the same company.

2.7.1   Email Account Creation

The following quick creation set-up steps will suffice for most organizations; please follow these steps and if you are not able to successfully test your email account upon completion, please follow the more detailed steps under "Account Creation".

If you are using a web-based email account, it may need to be configured to "allow access for less secure applications." This is done to allow KnowledgeSync or other applications the ability relay mail through the email provider.

- Go to KnowledgeSync **Admin > Services > Email Accounts.**
- Click the **Add** button located at the top left of the window. The Email Account Editor module opens.
- Click the appropriate Email Type: SMTP or OAUTH2.

2.7.1.1   SMTP Email Type

At the bottom of the module, select which email defaults to apply:
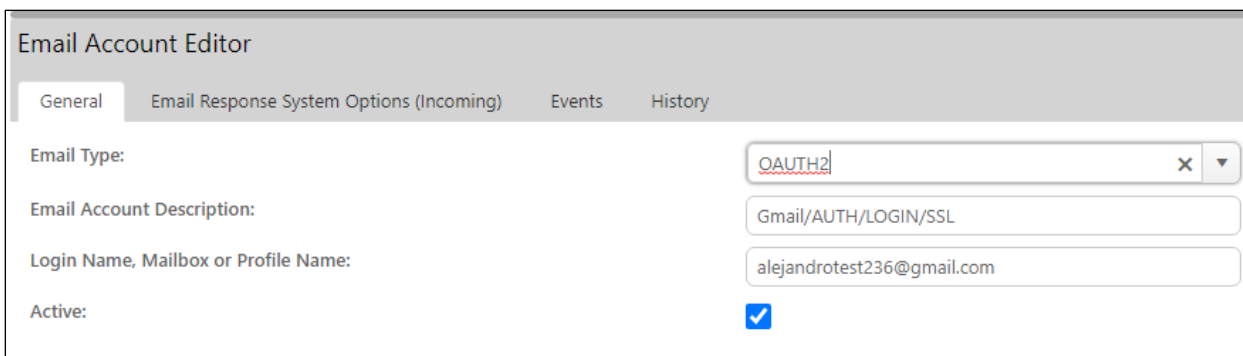
- o   Gmail
- o   Hotmail
- o   Yahoo
- o   Exchange
- o   Outlook 365

In the **Email Account Description** field, you may change the text BEFORE the first slash to reflect a description of the account, but it is critically important not to delete or change the remainder of the information.

- In the **Login Name, Mailbox or Profile Name** field, enter your user ID. This ID is the full email address account name, for example: MyAccount@gmail.com.
- In the **Password** field, enter the password for the email account.
- Ensure the **Active** box is checked.
- **TLS and Authentication Methods:** For the TLS (Transport Layer Security) and Authentication methods, do not change the defaults.
- In the *SMTP/IMAP Internet Email Information* section, enter in the following information:
  - Outgoing Mail (SMTP) Server Name
  - SMTP From Name (Required for Internet)
  - SMTP Port (Optional)
  - Incoming Mail (IMAP) Server Name
  - IMAP Port (Optional)
- Click **Save**.

2.7.1.2   OAUTH2 Email Type

Please note, if using OAUTH2 email type, the on-prem installation relies on a service in the cloud to connect to Microsoft or Google. Users should use the Advanced Email Configuration for OAUTH2 for Microsoft or the KS Gmail OAuth Setup document for the client credentials workflow.

- **Select the Email Type.** Choose SMTP or OAuth2 from the dropdown menu.
- In the **Email Account Description** field, you may change the text BEFORE the first slash to reflect a description of the account, but it is critically important not to delete or change the remainder of the information.
- In the **Login Name, Mailbox or Profile Name** field, enter your user ID. This ID is the full email address account name, for example: MyAccount@gmail.com.
- Ensure the **Active Box** is checked.
- Click the button to Sign In button with Microsoft or Google and follow the prompts to finish setting up the email account.

2.7.2   Testing Email Accounts

- Click on the **Description** area of the email account you just created so the line becomes highlighted.
- Click the **Perform Email Test** button.
- Enter the desired recipient email address and click **OK**.
- Check the recipient email account to see if the test message has sent successfully.

Note: You may need to check the Monitor or Corrective Action area of Administrator if the send is not completed. You may also want to try re-starting the KnowledgeSync service if items are not sending but also not displaying an error in the Monitor.

2.7.3   Email Response System Options (Incoming)

This option refers to the ability for KnowledgeSync to monitor and respond to the content of incoming email messages, such as messages that are sent to an organization's generic email accounts – e.g., info@knowledgesync.com, support@knowledgesync.com, et cetera.

The configuration and use of the Email Response System (ERS) is covered in a separate chapter of this manual – please refer to that chapter for details.

2.7.4   Events tab

On this tab, all events are listed with their associated applications. A green check mark indicates the event is active.

To update a sender account for an event, check the box in the **Is Sender?** column and then click the **Update** button to set the current email account as the sender on all events.

> Note: If the current email is already the sender account on an event, the checked box for the *Is Sender?* column is disabled. This prevents removing the sender account from the application. Only users with permission to edit or create events or to create an events email alert can make changes to the event definition tab.

2.7.5   History tab

Lists the following information about the email account:

- Created By
- Created On
- Last Changed By
- Last Changed On

2.8   Webcast (Dashboard Alert) Locations

This option refers to the ability for KnowledgeSync to send alerts to dynamically published and updated web (HTML) pages. (E.g., a person in accounting might want a dynamic web page that shows all invoices due this week, while someone in shipping might want a dynamic web page that shows a list of all orders due to ship today.)

Please refer to the chapter titled "Dashboard Alerts (Webcasting)" for complete details on the configuration and use of this module.

2.9   SSRS Locations

KnowledgeSync triggers, generates, and delivers reports – either on a "scheduled" basis (e.g., an hourly stock status report), or on a "triggered" basis (e.g., an A/R report that's generated only when a client exceeds their credit limit.). You may use either Crystal Reports or MS SQL Server Reporting Services (SSRS) with KnowledgeSync to generate and deliver your reports.

For SSRS reports only, you need to specify some initial configuration details. Expand the "Admin" module and under "Services", click on "SSRS Locations." When you add or update an SSRS location, the following window opens:

Fill in the following fields:

- **Description**: The name of the SSRS server (e.g., "SSRS on QA-SVR123")
- **Active**: Whether this SSRS configuration will be available to KS events.
- **Report Server URL**: This is the URL of the SSRS Report Execution 2005 Service on the report server (e.g., "http://qa-svr123/ReportServer"). Copy this URL from SQL Server Reporting Services Configuration Manager.

  Make sure the URL is accessible from the machine where KnowledgeSync (the web application and Windows Service) is running. Use a "localhost" URL only if KnowledgeSync and SSRS are installed and running on the same machine.

- **Report Server 2010 URL**: This is the URL of the Report Service 2010 Service on the report server.
- **Notes**: Use this field to contain comments about the corresponding SSRS location.
- **Username/Password/Domain**: This is where you specify the credentials used to run the Report Execution service of SSRS. The values in these fields must match the settings in the SQL Server Reporting Services Configuration Manager.

The domain can be entered separately (in the "Domain" field) or it may be included in the username, as in "mgmt\svcQASQL".

2.10 SMS Accounts and Twilio Integration

One of the ways that KnowledgeSync can send alerts is via text messaging to a mobile phone, generally referred to as SMS or "short message service".

KnowledgeSync uses Twilio as its SMS Provider and users will need to purchase a Twilio subscription to send text-based alerts. You will need to enter two fields (account SID and auth token) from Twilio into KnowledgeSync.

Twilio offers many service options. You will need "Programmable SMS" for sending texts using KnowledgeSync.

To configure KnowledgeSync to send SMS alerts, open the Administrator, expand the "Services" option, and select "SMS Accounts". Click on "Add" to create a new SMS account and the following window will appear:
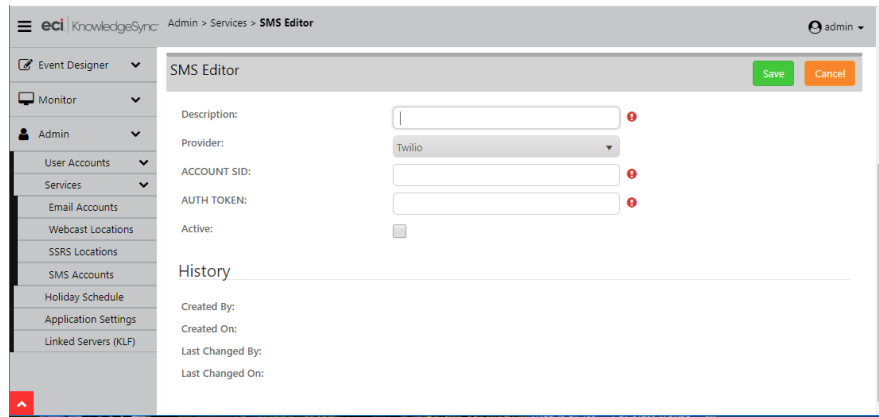
SMS (ALERTS VIA TEXT MESSAGE) SETUP

Fill in the fields as follows:

Description: The name of the SMS service (e.g., "KS Texts")
Provider: Select your choice of testing provider (currently, "Twilio").
Account SID: (you receive this from the texting provider)
Authorization Token: (you receive this from the texting provider)
Active: Click in this box to enable this SMS account for use in sending alerts via text message.

Click on the "Save" (or "Update") button to retain your entries.

2.11 Exact Synergy Integration

Exact Synergy is a request and task management application that is frequently used in combination with KnowledgeSync. Typically, Synergy integration with KS is used so that when KS detects an important business condition, it can trigger the creation of a request/task within the Synergy application. It's also worth noting that KnowledgeSync can monitor the Synergy application for critical, time-sensitive conditions and can trigger corresponding alerts and workflow.

To configure KnowledgeSync's integration with Synergy, open the Administrator, expand the "Services" option, and select "Synergy Servers". Click on "Add" to create the integration and the following window will appear:
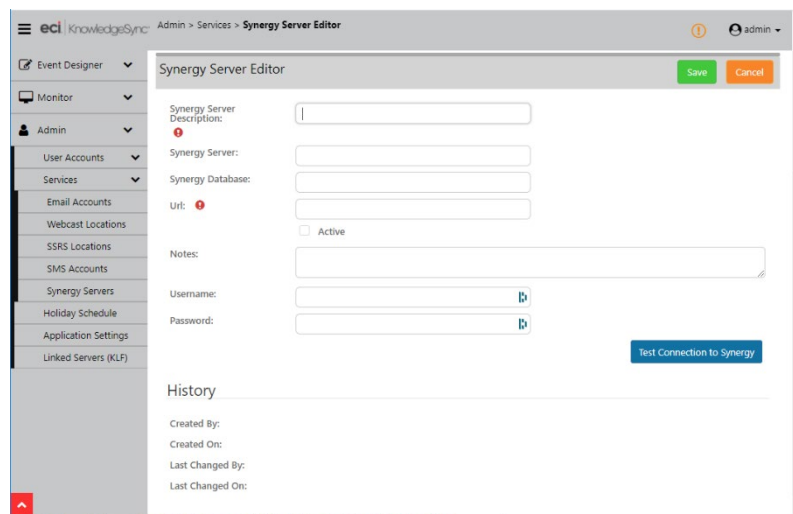
Fill in the fields as follows:

- **Server Description**: The name of the Synergy server being connected to KS
- **Server**: The location of the Synergy server . . .
- **Database**: The location of the Synergy database . . .
- **URL**: The URL address of the Synergy . . .
- **Notes**: Details about this Synergy server
- **Username**: The user account used to log into the Synergy server
- **Password**: The password for the account specified previously

Click on the "Save" (or "Update") button to retain your entries. Be sure to confirm a successful connection to your Synergy server by using the button called "Test Connection to Synergy".

2.12 Holiday Schedule

You can control whether a KnowledgeSync event will run if it falls on a holiday. To define your holidays (and review pre-configured holidays), expand the Admin module, and then click on "Holiday Schedule". You may either add a new holiday or review and edit existing holidays.

To edit the configuration of an existing holiday, click on the blue "Edit" button next to the desired holiday in your window.

For each holiday, you have access to the following data:

Holiday description (e.g., "Christmas Day")
Holiday date (you can select from the calendar icon)
Whether the holiday falls on the **same date every year** (e.g., Christmas always falls on December 25th)
Whether the holiday is currently **active**, thus enabling an event to be configured to observe that holiday and not be submitted on that date.

> *Each event may be configured to either run or not run on a holiday.*

HOLIDAY DEFINITION

2.13 Application Settings

KnowledgeSync has several system settings that help you customize the application to your specific needs and enable & disable diagnostic settings in case you need more details on the execution of events (such as for debugging an event that is failing). You can use the "Application Settings" option under the "Admin" menu option or click on the "gear" icon in the top-right of the KnowledgeSync window.

There are 4 types of system settings:

Portal: These control inactivity, log file and backup settings for the web portal.
Engine: These are additional log and backup settings, as well as the location of email alert attachments and the location of KnowledgeSync files.
Engine Modules: These control the diagnostic "level" (degree of detail) settings for the individual KnowledgeSync servers. Note this is also where you can specify the maximum "retries" for when KnowledgeSync is unable to send alerts via email, SMS, and webcast alerts.
Automated Cleanup: These control the automated removal of historical files from the KnowledgeSync system.

2.13.1 Error Log Reporting
Most of the options within KnowledgeSync can have "error reporting" turned on for enhanced details regarding any errors that KnowledgeSync is experiencing during event processing. The error logging level for the KnowledgeSync Portal is set in the "Portal" option within Application Settings; all other logging levels are set within the "Engine Modules" option.

Both the KS Portal and KS Service write text-based file logs on the server that can be used for troubleshooting any issues that might occur.

2.13.2 Number of Log Files to Retain
The number of backup log files retained is determined by the setting "**Maximum Number of Backup Log Files to Keep**" (in both the "Portal" and "Engine" tabs). Set this value to '1001' to retain all log files.

2.13.3 Log File "Rolling Style"
In both the Portal and Engine tabs, there is a setting for the "**Log Based on Rolling Style**" The "**rolling style**" determines how and when new log files are created.

Once – Each module has a single log file which will grow as data is added to it. The logs will be backed-up and replaced by a new file only when the KnowledgeSync Windows Service is stopped or when the KnowledgeSync web application in IIS is stopped and restarted. Older log files will have names such as KSService.1.log, KSService.2.log, et cetera.

Size – Each module creates a log file that is backed-up and replaced by a new file when the current log reaches the size configured in "**Log File Maximum Size**". Older log files will have names such as KSService.1.log, KSService.2.log, et cetera.

Date – Log files are backed-up and replaced every day. The format of the log file name is determined based on the settings of the "**Log Date Pattern**" and the "**Log File Path**" (see following sections). As an example, a portal log file from 10/1/2020 would have the name PubKSPortal-20201001.log

Composite – This is a combination of **Size** and **Date** styles described previously. A new log file is created when the size limit is reached or the date changes.

2.13.4 Log File "Date Pattern"
The "**Log Date Pattern**" (set in both the Portal and Engine tabs) is the format of the date (for the file name) when used in the naming of log files.

2.13.5 Log File Paths & Names
KnowledgeSync uses the following settings to determine the locations and names of its log files:

Portal: See the setting "**Portal Log File Path**". The default name and location is:

C:\ProgramData\KnowledgeSync\KSLogs\PortalLogs\PubKSPortal.log

Engine: See the setting "**Service Log File Path**". The default name and location is:

C:\ProgramData\KnowledgeSync\KSLogs\ServiceLogs\KSService.log

2.13.6 Log File Names for the Engine Modules

Within the "**Engine**" tab is a setting called "**Individual Log Files for Engine Modules**".

This setting determines whether the Engine modules listed below will write a single log file for all the engine modules (set to 'No'), or will write separate log files for each engine module (set to 'Yes').

If the "Individual Log Files for Engine Modules" setting is enabled, the following log files are created for the engine modules:

- KSService-Action.log

- KSService-ApplicationEvents.log

- KSService-Copy.log

- KSService-Email.log

- KSService-ERS.log

- KSService-Report.log

- KSService-Scheduler.log

- KSService-SMS.log

- KSService-Webcast.log

2.13.7 Logging Levels

Set the logging levels as follows:

- ERROR: Only exceptions and errors that occur in the application will be logged. This is the default (and minimum) level of error reporting.
- WARN: Errors and warnings are logged.
- INFO: Informational messages are written to the error logs, in addition to errors and warnings. These include such items as parameters passed into a query or report.
- DEBUG: This is the most detailed level of log reporting. Debug level diagnostics are written in addition to INFO, ERROR and WARN messages. These include query results, email message texts, and other detailed data.

In general, we recommend setting the log level to ERROR or WARN during normal processing.

To modify any of these settings, simply click in the corresponding "Value" field and type in your selection. Those settings which have pre-defined values to choose from (such as a log level of "info", "warn", "error", and "debug") and will show a list button when you click in the "Value" field.

2.13.8 Engine Modules: "Maximum Attempts"

If KnowledgeSync attempts to run an event and an error occurs, KnowledgeSync will attempt to "re-try" running the specific event component that failed. The number of times KnowledgeSync will attempt to re-try the failed process is referred to as "maximum attempts". You can set the number of maximum attempts for the following event components in the "Engine Modules" tab:

- "Events": Event triggers
- "Email": Sending an email alert
- "SMS": Sending a text (SMS) alert
- "Webcast": Sending a webcast alerts
- "Reports": Generating an event's reports and/or charts
- "Actions": Executing an event's "actions"

The "Maximum Attempts for . . . "setting specifies how many times an operation is retried before the item is moved to the "Error" category in the KS monitor.

Note that the "Error" display options in the monitor each have a "**Retry**" button – and clicking on this button will reset the number of "maximum attempts" to zero for the item, and thus allow the module to attempt to process the item again.

2.13.9 File Paths for KS EventPaks, Reports, Scripts & Other Objects

In addition to log file paths, several other KnowledgeSync file paths can be updated through Application Settings:

In the "**Portal**" Tab:

- **Epak File Path**: Where eventpak files are built on the server before being downloaded to the client machine.
- **Reports File Path**: Where generated reports (Crystal & SSRS) are saved on the server.
- **Scripts Path**: Where Visual Basic script files are stored on the server.

In the "**Engine**" tab:

- **Attachments Path**: Where an event's file attachments are generated / stored on the server before they are attached to emails or webcasts.
- **Email Response Attachments Path**: If KS is configured to monitor incoming email messages, this location is where received email attachments are saved on the server.
- **Service Installed Path**: This an information value only; it Indicates the path on the server where the KS Service is installed.

2.13.10   Automated Cleanup

The "**Automated Cleanup**" tab (within application settings) is used to schedule a job that removes the history of "completed" events from KnowledgeSync. The options are:

Scheduled Time: The time of day (in AM/PM format) when this job runs. The default is 1:03 AM.

Keep Records for (Days): The number of days of completed event history data to be retained in the KS database.

Schedule This Task: Whether the automated cleanup job is active. Un-checking this option means that the cleanup job will not run.

Please note that there is no option to do an ad-hoc submission of the automated cleanup job. However, "Schedule This Task" setting can be temporarily checked on to enable a single run of the job, after which the setting can be unchecked.

2.14 Linked Servers

If you are integrating KnowledgeSync with a Sage 100 Providex database environment, please use the stand-alone KnowledgeSync Linking Function (KLF) that is located on the "Downloads" page of the Sage Alerts & Workflow website instead of the "Linked Servers" option. The KLF program and documentation are located on the downloads page.

KnowledgeSync can monitor information in a database via .NET database Provider, OLE/DB, ODBC, and REST API web service.

For some non-SQL databases, you'll find it best to create a "link" between the target database and SQL Server. This is called a "linked server" and it optimizes retrieval and analysis for non-SQL databases that do not have a strong native database driver. A linked server has the following benefits:

- It reduces the amount of time to retrieve data.
- It enables calculations not supported by the database's native driver.
- It creates single-field unique "keys" for each database table.

The process of using a linked server for a database is as follows:

1. You create a 64-bit ODBC source that points at your target database.
2. You use KnowledgeSync to create a linked Microsoft SQL Server database.
3. You connect KnowledgeSync to the linked database using the SQL database Provider.

2.14.1 Create a Linked Server

After you have created a 64-bit ODBC source that points at your target database, the next step is to create the Linked Server using the ODBC connection you created. Note that the linked server and database get created on the same SQL Server where KnowledgeSync's database tables reside.

1. From the KS "Admin" option, select "Linked Servers"
2. Click on the "Add" button.
3. Enter a name for the linked database.
4. Select the ODBC source that points at your target database. Note that the list button for this field does not display the available 64-bit sources; you must manually type in the name of the 64-bit source.
5. Click on the "Create Linked Server" button.
6. Once you receive the message that the linked server has been created, click on the "Tables" tab within the Linked Server option. You'll see a list of all the tables in the target database.
7. Although you can select specific tables for the linked server, we recommend checking the box titled "Link Checked Tables" to select ALL tables.
8. If your target database has multiple "entities" (e.g., companies), go to the field titled "Create an optional change tracking column" and type in the entity/company code of your first entity. (You'll need to create one linked server per entity.)
9. Click on the "Link" button at the top of this window.

Any tables that are not able to be linked will be detailed in the box at the bottom of the Linked Servers option.

The linking process may take some time to complete (e.g., 10 minutes or more) depending on the size of your target database.

2.15 The KnowledgeSync On-Line Knowledgebase

If you have questions about an aspect of KnowledgeSync, it is suggested as a first step that you scan the on-line knowledgebase of frequently asked questions (and answers). You can access the knowledgebase from:

https://www.knowledgesync.com/downloads/documentation-assets

2.16 Contacting KnowledgeSync Technical Support

If you are unable to find a solution to your issue via the knowledgebase, the best way to access KnowledgeSync Technical Support is by calling 1-800-850-8055 or sending an email the Customer Support team.

3   Query Designing

3.1   Query Designing

A KnowledgeSync event consists of two main parts:

- The triggering condition
- The response to that condition

Events may be triggered by:

- A database query
- A REST web service API (detailed in the "Web Service Integration" chapter)
- A Visual Basic (VB) script (detailed in the "Visual Basic Scripting" chapter)

Database queries can identify business conditions by using:

- A .NET Data Provider
- An ODBC data source
- An OLE/DB data source

This chapter covers the design and use of <u>database queries</u>.

A query determines whether specific conditions exist within an application's database. An event uses one (or more) queries and if the conditions of its queries are met, the event is "triggered" and the resulting alerts, reports, and workflow actions are executed.

Most events use a single query; some events use multiple queries, especially events whose triggering is based on a combination of conditions that exist across multiple application databases.

There are three types of queries. The first is a **<u>record level</u>** query which checks for records that meet certain conditions. For example:

- Invoices that are overdue for payment
- Stock that is approaching its re-order level
- Sales quotes that expire in 3 days

The second is an **<u>aggregate</u>** query, and it checks to see if a <u>group</u> of records <u>collectively</u> meet a certain condition. For example:

- A <u>total</u> of past due invoices for a single client that exceeds $10,000
- A <u>count</u> of more than 3 late deliveries from a supplier over the last 6 months
- An <u>average</u> value of open quotes that is less than $5,000

The third is a **<u>field-level change</u>** query, and it checks to see if one or more fields have had their value <u>changed</u>. For example:

- Customers whose credit status has changed
- A support call whose priority has changed to "Urgent"
- An inventory item whose list price has changed

3.2   One Query in Multiple Events

Consider an organization that wishes to receive the following alerts:

- Clients with overdue invoices > $2,000
- Clients with overdue invoices > $5,000
- Clients with overdue invoices > $10,000

These three alerts <u>could</u> each use their own query; each query checking for the corresponding dollar threshold.

However, KnowledgeSync allows <u>one query to be used in multiple events</u>; a query in which the triggering dollar threshold is specified on the <u>event</u> level.

3.3   Preparation

You'll find query design easier if you begin by identifying the **<u>event conditions</u>** that you wish to monitor.

A query serves two purposes; it identifies what <u>conditions</u> must be met for an event to be triggered, and it <u>identifies which database fields</u> may be used within the event's alert messages and workflow. An event uses database fields in 4 ways:

As content for its alert messages.
To contain an alert recipient's delivery address (e.g., a field that contains a customer's email address).
As selection criteria to be passed into a Crystal or SSRS report.
As content for an event's workflow actions (such as data to be written into an application's database).

3.3.1   Record-Level Queries

The first step in any query is to identify its purpose. For example:

> **"When a 'critical' service call is logged, send an email alert to the customer's account manager."**

Consider what you'd like the alert message to look like. For example:

```
Hello, Don Farber:

The following critical support call was just logged for client Acme Incorporated:

Contact:  Tom Jones
Call Received:  09/14/2001  1:35 PM
Type of Problem:  Fatal error message
Status:  Open
Assigned to:  Lisa MacKinnon
Description:  Error 1094 during login.
```

You should identify the <u>data fields</u> you wish to put into the alert:

```
Hello, [salesrep_name]:

The following critical support call was just logged for [customer_company]:
Contact:  [customer_name]
Call Received:  [ticket_log_date]  [ticket_log_time]
Type of Problem:  [ticket_type]
Status:  [ticket_status]
Assigned to:  [ticket_rep_name]
Description:  [ticket_description]
```

By taking these steps, you'll know what fields the query needs to retrieve data from. And by reviewing the event's description ("... *when a 'critical' service call is logged*") we can identify the <u>conditions</u> that the query needs to check for:

Calls with a priority of 'critical'
Calls that are received today

In terms of the recipient of an alert, in some cases, recipients are "hard-coded"; e.g., notifying your CEO or CFO. In other cases, you might wish to notify a person who is **<u>associated</u>** with the record being triggered as in the following example:

"Send an alert <u>to the customer's account manager</u> when a 'critical' service call is logged."

In this scenario, the alert recipient will depend on the client for whom the critical call was logged. If you look back on the alert message, you can see that the person who should receive the alert message is "[salesrep_name]".

However, "[salesrep_name]" is not an email address; if you look further into the salesrep data, you might find a field called "[salesrep_mail_address]" – and <u>this</u> is the field that contains the address that you wish to send the alert to.

If an event will send alerts to address(es) derived from the triggered record, the query must include the field(s) that contains those address(es).

KnowledgeSync events can also generate & deliver forms & documents (such as invoices or statements) as well as reports.

If an event will generate a report, review the report to see if it uses <u>parameters</u> (selection criteria). If it does, the query must retrieve the data which will get used as (passed into) the parameters. For example, an event might trigger an A/R Aging Report for an overdue client. If your report has a parameter on "client ID", you need to make sure that your query retrieves the data from the "client ID" field, so that value can be automatically passed into the report's parameter.

The last preparatory step is to identify whether an event will execute workflow to add or update records in an application database. If so, your query must include the data fields required for use by the workflow update procedures.

3.3.2   Aggregate Queries

Aggregate queries are designed much in the same way as record-level queries except that aggregate queries will use one of the following 5 options:

- Count (e.g., fewer than 5 assigned leads)
- Summarize (e.g., less than $10,000 in sales)
- Average (e.g., average response time is 47 minutes)
- Minimum (e.g., the first order placed by a client)
- Maximum (e.g., the most recent delivery)

Aggregate queries should be designed with the same considerations as record-level queries except for the fact that an aggregate query is going to be triggered by a condition related to a "group" of matching records. For example:

> **"If a client has more than 3 overdue invoices, send an email alert to the client's account manager, attach an "Open Receivables" report for that client, and schedule a phone call for the account manager to contact the client."**

And – correspondingly – the alert message might look like this:

> **The following client has more than 3 overdue invoices:**
>
> **Client: Jim Ramey**
> **Number of Overdue Invoices: 5**
> **Total Value of Overdue Invoices: $27,960.28**

3.4   Design Basics

For every query you should identify the following:

- Purpose of query
- Alert message
- Application tables which contain the data to be retrieved
- Fields of data required by the query
- Triggering conditions
- Alert recipients

3.5   Before You Design a Query . . .

Before you start working with queries, you must have created one or more "applications" in KnowledgeSync – either by downloading and installing eventpaks, or by manually creating applications that are linked to an underlying database or other source of data.

See chapter 1 ("Step #1: Connect KnowledgeSync to Your Application(s)") for more details on the ways you can link KnowledgeSync applications with underlying databases (such as .NET Database Provider, ODBC or OLE/DB).

3.6   Add/Edit a Query

To add a query, log into KnowledgeSync, click on the **Event Designer** option on the left and then click on "Application Events." (You'll see a list of applications currently configured in KnowledgeSync.)

In the "Applications" window that appears, click on the arrow to the left of the application you wish to work in. This will expand the selected application as shown in the following window:
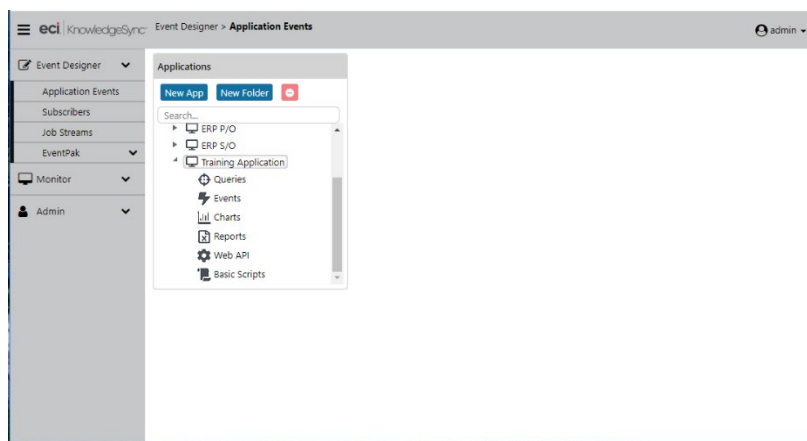
A SELECTED APPLICATION'S COMPONENTS

Click on the "Queries" branch and you'll see a window like the following:
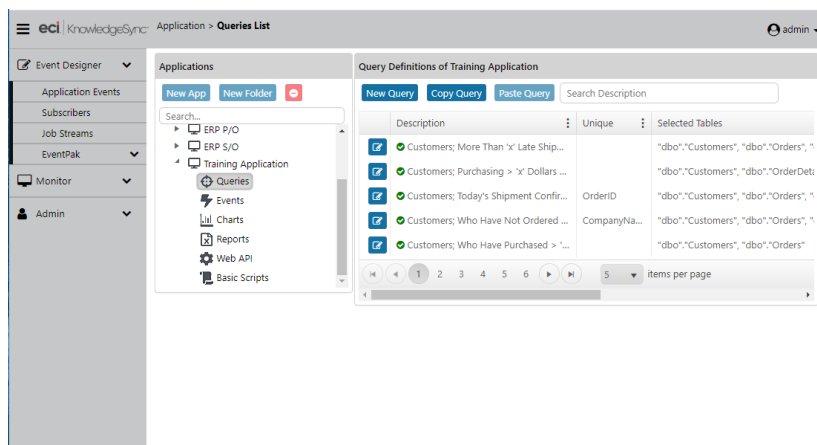
AN APPLICATION'S QUERIES

To add a query, click on "New Query" in the Query Definitions window.

You can use the "Copy" and "Paste" buttons to copy a query, and you may also use the "Search" field to locate a specific query based on the query's name.

To view or edit a query, click on the blue "Edit" button to the left of a query's description; to delete a query, click on the red "Minus" sign at the extreme right of the query (you may need to scroll the Query Definitions window to the right).

The following sections guide you through <u>record-level</u> queries as well as <u>field-level change</u> queries. Following these instructions is a separate section on the creation of <u>aggregate queries</u>. That section goes over only the *differences* between record-level and aggregate queries, so please review the record-level query section to become familiar with the basics of query design.

3.6.1   Description

The "Query Description" field contains the name of your query. Make your names short, consistent, and descriptive. We recommend starting the description with the item being monitored, followed by the condition. For example:

      Jobs; Shipped Today

The "Display Description" will default to the same value as the query's description.

It's the <u>display</u> description that will show in the list of queries, and that it's the display description that will be used when you <u>search</u> to locate a specific query.

Click "Active" to enable a query to be used in an event.
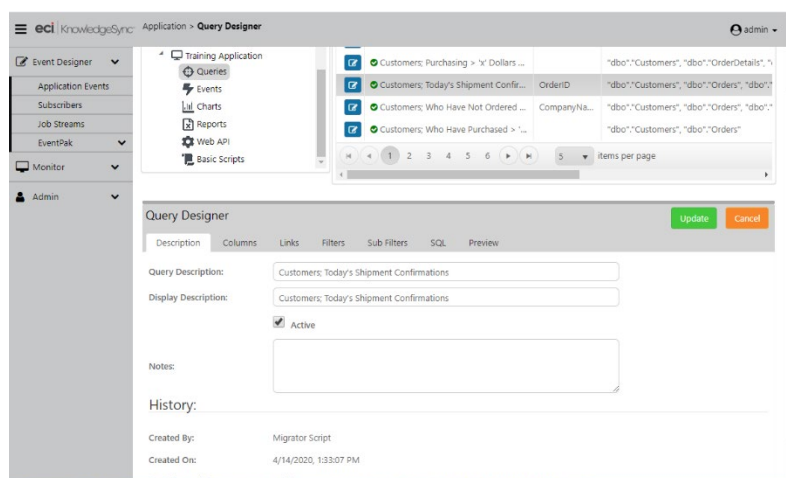
QUERY DESCRIPTION

3.6.2   Tables & Columns

The **Columns** tab is where you choose what database tables and views (and what columns within those tables/views) a query will retrieve data from. There are four things you need to consider:

What data might you wish to put into an alert message?

Even a query as simple as "jobs shipped today" would probably need to retrieve data from more than just the "job" table. The corresponding client data might be stored in an "account" table, and the client's account manager's details might be stored in a "salesrep" table.

Where are the alert addresses of the person(s) you wish to notify stored?

If you wish to notify a person who is "associated" with a triggered record, you need to include the table that contains their delivery address (such as their email address).

Will an event that uses the query execute any workflow actions (e.g., will it need to add or update information in an underlying database)?

Since an event can execute workflow to add or update database records, you need to consider what data the workflow needs access to.

Will an event that uses this query also execute a Crystal or SSRS report?

If an event is going to execute a report, see if the report uses "parameters" – as those parameter values will have to come from the event's query.

To select columns for a query, you first need to select the table(s) that contain those columns. Go to the list of <u>Available Tables</u> and click on green "Plus" sign to the left of each desired table. (You may also double-click on the desired table name.)

If you accidentally select the wrong table for a query and wish to remove it, go to the list of <u>Selected Tables</u> and click on the red "Minus" sign next to the table you wish to remove.

3.6.2.1   *Choosing the Same Table Multiple Times (Table Alias)*

Consider the following scenario: You want an alert about invoices that are overdue. Each invoice is associated with a customer and a salesrep. A table called "email" stores the email addresses of <u>both</u> the client and of the salesrep.

In order to get <u>both</u> email addresses, KnowledgeSync needs to access the "email" table <u>twice</u> – once for the customer's address, and once for the salesrep's address. But – in SQL -- you cannot choose the same table <u>twice</u> in a query, so KnowledgeSync lets you create a "table alias" for just such occasions.

When you select the same table multiple times, you are given the ability to specify an <u>alias name</u> for that table.

3.6.3   Columns

After you have selected the tables for a query, the next step is to select the columns (data fields) that you might wish to use in your event. Consider the following when selecting columns:

1) What data do you want to put into your alert messages?
2) Is the address of a person you wish to notify contained within a database field?
3) Will your event need to pass data into a report's selection criteria?
4) Will your event need to use data in an application update (workflow)?
5) Will your event need to "sort" matching records by certain fields?

3.6.3.1   *How to View & Select a Table's Columns*

Under "Selected Tables", click on the blue right-arrow to the right of the table whose columns you wish to display (or double-click on the desired table). The columns of the selected table will appear.

To add a column to a query, go to the list titled "Columns of Selected Table" and either click on the green "Plus" sign to the left of the column name or double-click on the desired column.

> *In addition to the column's name, you are also shown the <u>type</u> of data stored in the column (e.g., text, date, numeric, etc.) and the length of that column.*

As you select columns for a query, you may need to move to ensuing pages to see all your selected columns – which will look like the following:
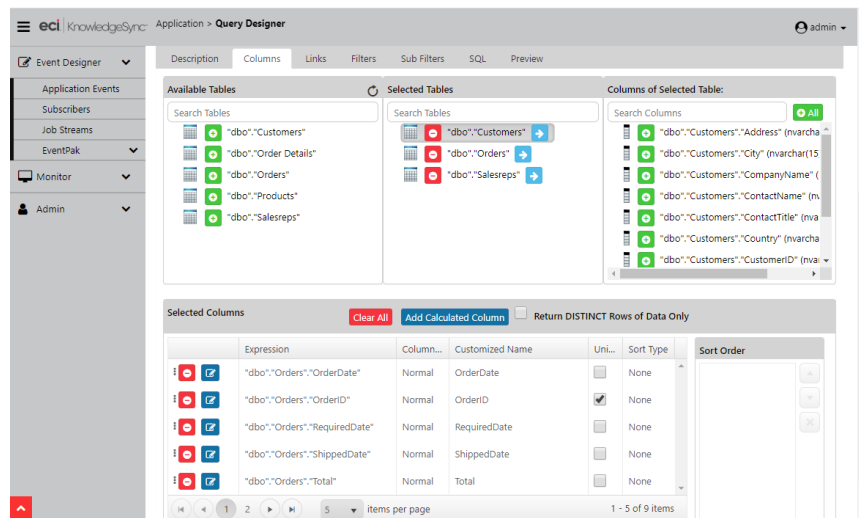
COLUMNS SELECTED FOR A QUERY

3.6.3.2   Selecting All Columns

To include all columns for a selected table, click on the green +All button.

Although this is an excellent way to <u>see</u> all data from a specific table, this is <u>not</u> suggested for normal query building as it makes it more difficult to see (and select) only those columns needed for an event.

3.6.3.3   Editing/Removing a Selected Column

To edit a column (to include calculations, change its format, etc.), go to your list of Selected Columns and click on the blue "Edit" button to the left of the selected column's name.

You will be taken into a special editing window, complete with advanced calculation functions to choose from.

To remove a selected column from a query, click on the red "Minus" sign to the left of the selected column's name.

3.6.3.4   Column "Type"

Record-level and field-level change queries always use a column type of "Normal". The other column types are detailed under the "Aggregate Queries" section of this chapter.

3.6.3.5   "Customized Name"

When an event's alert message is designed, you refer to the query's columns by using the columns' "Customized Names." This is important, as some applications have cryptic names for their columns. For example, an "activity" table may have a column called "ddate" – designed to store an activity's "due date".

Using a column's "Customized Name", you can create your own "label" for a field, for example "due_date" for the scenario above.

When you select a column for a query, KnowledgeSync will take the column's name and place it in the "Customized Name" field. You can then modify that value.

Here are some notes about using customized names:

- Do not include blank spaces. (use "due_date" and not "due date").
- Keep customized names under 25 characters.
- Do not begin a customized name with SQL reserved words, such as "sum", "sub", "add", "avg", "min", "max", "total", "trim", and "format".
- Do not have multiple fields with the same customized names; you won't be able to tell them apart.
- Use only letters and numbers in a customized name; do not use symbols.

- Do **not** change the default Customized Name of a query's "unique" column if the query tracks <u>changes</u> to the value of one or more database fields.

3.6.3.6   A Query's "Unique" Column

In KnowledgeSync, you can specify whether an event will trigger just <u>once</u> for a record that meets its criteria, or will trigger <u>repeatedly</u>, each time the event runs, for as long as a record continues to meet the event's criteria. For example:

Trigger <u>Once</u>: At 9 AM, activity #123 is overdue and triggers an alert. At 10 AM the activity is still overdue, but since it already triggered, it does not trigger another alert.
Trigger <u>Repeatedly</u>: At 9 AM, activity #123 is overdue and triggers an alert. At 10 AM – and every hour thereafter, it triggers and alerts again (until the activity is no longer overdue).

Although the option for one-time versus repeat triggering is specified on the <u>event</u> level, you need to do something on the <u>query</u> level to <u>enable this option</u>.

To enable records to trigger only <u>once</u>, KnowledgeSync has to "remember" a triggered record. This is done by identifying which column uniquely identifies a triggered record. (This is often referred to as a record's "key"; in KnowledgeSync this is referred to as a query's "unique" column.)

If you do **not** specify a query's unique column, any event that uses this query will automatically be in "repeat" mode. If you **do** specify a query's unique column, you will have the <u>option</u> (on the event level) to specify one-time or repeat notifications.

To specify a query's unique column, go to your list of selected columns for a query and put a checkmark in the "Unique" box of the **one column** that uniquely identifies each record (such as a "client ID" or an "order number").

Notes:

Always specify a unique field for non-aggregate queries.
Only <u>one</u> column may be checked as unique. If a unique is based on the combined values of two or more fields (e.g., "order number" and "line number"), create a multi-field unique (see "Multi-Field Unique").
Do **not** change the default Customized Name of a query's unique column if the query tracks <u>changes</u> to the value of one or more database fields.
The maximum length of a unique column for a query is 64 characters.

3.6.3.7   How a Record Becomes "Un-Triggered"

Consider an event that runs daily and is set to "not repeat" its alerts about customers who get placed on credit hold.

On Monday, Acme Corp is placed on credit hold and an alert is sent.
On Tuesday, Acme Corp is still on hold. The event (set to "not repeat") runs but does **not** trigger another alert about Acme.
On Wednesday, Acme is taken off of hold. The event runs, sees that Acme no longer meets the event's criteria, and "<u>clears" ("un-triggers") the Acme record</u>.
On Thursday, Acme is once again placed on credit hold. The event runs, is newly triggered, and another alert is sent.

3.6.3.8   Multi-Field Unique

To create a multi-field unique, you'll need to concatenate (combine) the values from the two (or more) fields.

The "type" of fields you concatenate must be "character" or "string" type fields. To concatenate fields of other types (e.g., numeric or date), you must <u>convert</u> the fields into character format first. (See "Field Editing Notes".)

To concatenate fields (for SQL Server), follow these steps:

Click on the button called "Add Calculated Column".
Choose the first database column that you wish to comprise your concatenated unique.
Click in the Calculated Column Workbox and enter a plus sign (+) after the column name you have selected.

Choose the second database column that you wish to comprise your concatenated unique. The concatenated column is now complete and looks like the following:

```
sysdba.orders.order_number+sysbda.orders.order_lineno
```

(The concatenation symbol for an Oracle database is two vertical bars (||).)

> *You cannot use a concatenated unique if your query is checking for a "change" to the value of a field in a database. (Instead, create a "view" to the desired table and concatenate the unique columns within the view.)*

3.6.3.9   Order Line Item Unique

In many ERP systems, an order's details consists of a "header" and "line item details". Because of this, every order detail line has the same order number associated with it.

In such cases, make sure that the query's unique consists of both the order number AND the line number. That way, each detail line has its own unique value.

> *You can concatenate only text-based fields; thus if you needed to concatenate the "order.number" and "order.lineno" column but both are numeric in format, your concatenation syntax (under SQLServer) would appear as follows:*
>
> *convert(char(10),order.number)+convert(char(10),order.lineno)*

3.6.3.10   Calculated Fields

Examples of calculated fields are:

- The number of days until an invoice is due
- Unit sales price times quantity ordered
- Percentage of credit limit remaining

Calculated fields may be used in a query as well as in an event's alert messages, report selection criteria and workflow actions.

To create a calculated field, click on the "Add Calculated Column" button. To add a calculation to a selected column, just click on the blue Edit button that appears to the left of that column.

The Calculated Column Editor provides you with a list of frequently-used arithmetic calculations at the top of the window; you may use any valid arithmetic symbols and functions (such as '+', '-', '/', and '*') as well as any valid SQL functions that are approved for use with the database and data provider you are using.

The editor also provides you with a list of all columns from your selected tables, as well as KnowledgeSync's complete list of date substitution variables. The "workbox" at the bottom of this window is where your calculation syntax is formatted.
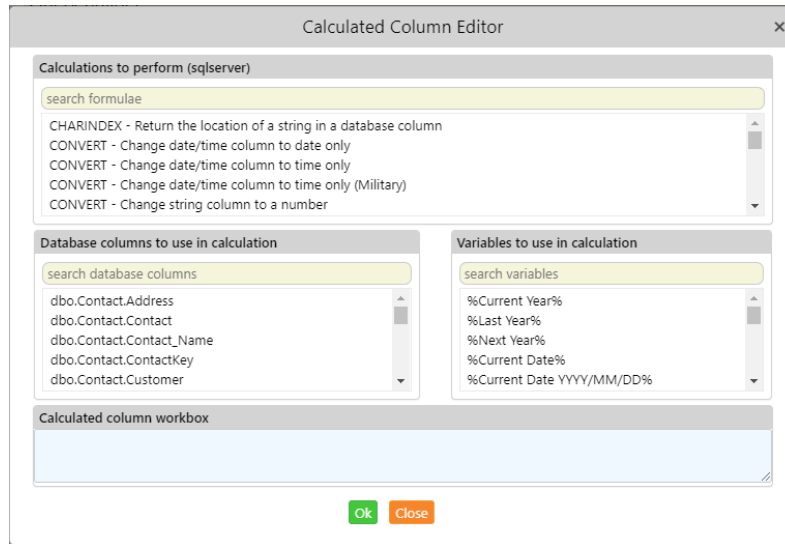
CALCULATED COLUMN EDITOR

If you select any of the functions from the list of calculations, KnowledgeSync will automatically format that function for you, and all you need to do is select any remaining elements for that function.

Example #1: a calculated field that computes the gross margin percent of an item.

The gross margin is calculated by determining the difference between an item's price and cost, multiplying that amount by one hundred, and then dividing that amount by the item's cost. You would do the following in the Calculated Column editor:

Double-click on the column that contains an item's price; you'll see something like:

`INVENTORY.ITEMPRICE`

In the workbox area, add a minus sign after that field

`INVENTORY.ITEMPRICE-`

Double-click on the column that contains the item's cost.

`INVENTORY.ITEMPRICE--INVENTORY.ITEMCOST`

Type in parentheses around this expression.

`(INVENTORY.ITEMPRICE--INVENTORY.ITEMCOST)`

Type in *100 at the end of your calculation.

`(INVENTORY.ITEMPRICE--INVENTORY.ITEMCOST)*100`

Type in an additional set of parentheses around the entire expression.

`((INVENTORY.ITEMPRICE--INVENTORY.ITEMCOST)*100)`

Type in the division symbol "/" at the end of your calculation

`((INVENTORY.ITEMPRICE--INVENTORY.ITEMCOST)*100)/`

Double-click on the item's "price" field again.

```
((INVENTORY.ITEMPRICE--INVENTORY.ITEMCOST)*100)/ INVENTORY.ITEMCOST
```

Click on "OK".

Example #2: A field that determines the number of days until an invoice is due.

In Microsoft SQL Server, this calculation uses of a function called "DATEDIFF", which calculates the number of days between two dates. Thus, you would click on the "Add Calculated Column to Query" button and choose the function "DATEDIFF – Difference in Days between two dates". The following will appear in the workbox area:

```
DATEDIFF(DAY,$Column$,$Column$)
```

Since you want to calculate the number of days between "today" and an invoice's due date, select the "%Current Date%" variable and follow that by selecting the column that contains an invoice's due date. Your result will look like this:

```
DATEDIFF(DAY,'{%Current Date%}',dbo.tarInvoice.DueDate)
```


3.6.3.11 Calculated Field Notes
Use a query's "Preview" to ensure that your calculation syntax is correct.
Be sure to modify the "Customized Name" of a calculated field.
A single query can have multiple calculated fields within it.
Copy and paste the contents of a calculated field by using CTRL/C to copy the contents and CTRL/V to paste them.
Verify the format in which an application stores its "date" or "date/time" values. Even though the values might be displayed in one format (e.g., mm/dd/yyyy), your app might store them in a different format (e.g., yyyymmdd).

3.6.3.12 Date-Related Functions
Most databases (and drivers) allow you to calculate the difference between two dates. The syntax of these functions is dependent on the database (and provider) you are using.

Here are a few examples:

Invoices Due Within 'x' Days (MS SQLServer)

```
datediff(Day,'{%Current Date%}',ARinvoice.DueDate)
```

Client Whose Last Order Was More Than 'x' Days Ago (MS Access)

```
datediff('d',"Invoice"."DateCreated",'{%Current Date%}')
```

Quotes Due to Expire Within 'x' Days (Pervasive SQL)

```
datediff(Day,oehdr.CancelDate,'{%Current Date%}')
```

The number of hours between a date/time field and the current date/time

```
datediff(hour,table.column,'{%Current Date%}'
```

Add 3 hours to the current date/time

```
dateadd(hour,3,'{%Current Date%}')
```

Add 3 hours to the value in a database field:

```
dateadd(hour,3,table.column)
```

3.6.3.13 Configuring a Query to Deliver to Multiple Dynamic Addresses

A common requirement in query design is when you wish an event to send alerts to <u>multiple</u> addresses that are drawn from database values. For example, an event monitoring open support tickets might need to notify both the rep whom the ticket is assigned to, and the salesperson who is assigned to that client's account.

Although an event supports multiple dynamic delivery addresses (such as the "send to", "cc", and "bcc" email addresses), only a <u>single</u> query field may be referenced for any one of those options (i.e., you can specify only <u>one</u> query column for the "send to" addresses).

In order specify <u>multiple</u> addresses for one of these options, you need to combine those addresses into a single field such as:

```
sysdba.ACCOUNT.EMAIL+','+sysdba.USERINFO.EMAIL
```

 (Multiple email addresses are separated with commas.)

3.6.3.14 Trimming "Padded Spaces" from a Value

Sometimes column values contain "leading" or "trailing" spaces and you may not want KnowledgeSync to include these spaces when it retrieves the column's value.

Depending on the database being monitored, you can use the following functions. (The following are for SQLServer databases.)

rtrim: Remove all following spaces from the right side of the field value
ltrim: Remove all leading spaces from the left side of the field value
trim: Remove all surrounding spaces from both sides of the field value

The format for using these functions is:

```
trim(table.column)    (SQLServer)
```

3.6.3.15 Retrieving 'x' Characters of a Value

Sometimes you might not want to retrieve the entire contents of a field; you may (for example) wish to retrieve only the day of a person's birthday (as opposed to the day, month, and year).

To do this, you would use the "substring" function as follows

```
Substring(table.column,starting position,length)
```

For example:

```
Substring(account.name,1,20)       (SQL Server)
```

This would retrieve the first 20 characters of an account's name.

3.6.3.16 Converting a Field's Data Type

If you need to convert a field from one data type (such as numeric) into another type (such as character) you would use the "convert" function as shown in the following example:

```
Convert(char(20),account.createdate,0)       (SQLServer)
```

This converts a date field into a string field.

For an Oracle database, the command is:

```
TO_CHAR(TABLE.NUMBER,'9999999')
```

This turns a numeric field into a string.

For a Pervasive SQL database, the command is:

      convert(ARTRAN.OrderNumber,sql_char)

This turns a numeric field into a string.

3.6.3.17 Retrieving the "Top x" Number of Records

Another useful function is the ability to retrieve only the top 'x' number of records that meet your criteria. (For example, the top five sales this week.) The syntax for doing so (under SQLServer) is:

```
Top x table.column
```

As in:

```
Top 10 sales.amount       (SQLServer)
```

3.6.3.18 Creating a Column that is a Label + Field Value

In some cases, you may wish to create a column that contains both a field "label" as well as the field's value. This is useful if you have a field whose label you do <u>not</u> wish to appear in an alert message if the value of the field itself is blank (or null).

For example, let's say that you have an ERP system in which you have four types of line items: inventory, non-inventory, miscellaneous charges, and comment lines. Only <u>inventory</u> items contain a warehouse code, and for those lines you wish to include a section in your alert message that says

```
WHSE:  Boston
```

(where "Boston" is the name of the warehouse from your ERP database)

Since the three other line types do not use a warehouse, you do not want the label "WHSE:" to appear in the alert message for those lines.

The answer is to <u>combine</u> the label ('WHSE:') with the value from the warehouse field. Thus your query column would look like the following:

```
'WHSE:'+'  '+InvDetail.Whse
```

And – as you'll learn in the "Event Designing" chapter, you can tell KnowledgeSync <u>not to include</u> (or allocate space to) any field whose value is null.

You can add labels to only <u>character-based</u> fields.

The preceding logic will work only if the lack of a value in the warehouse field is represented as a "null" value; this is because when you add a constant value (such as "WHSE:") to a column with a null value, the ending result is still a null value. (I.e., any value + a null value = a null value)

3.6.3.19 Creating a Column that Contains a Dynamic Date Value

KnowledgeSync includes many date substitution values, such as "current date", "last Monday date", "next Friday date", and so on. To use one of these fields as a column within a query, surround the variable with single quote symbols as in the following:

```
'{%Last Friday Date%}'
```

3.6.3.20 Retrieving Only the Month or Day from a Date Field

To retrieve just the month or day from a date field, use the following syntax:

```
month(hrpersnl.p_birth)              (SQLServer)
day(hrpersnl.p_birth)                (SQLServer)
```

3.6.3.21 Creating a Column that Contains a Hard-Coded Value

To create a column that contains a hard-coded value such as an email address (e.g., smith@KnowledgeSync.com), follow these steps:

1. Add any column to a query.
2. Edit the column you just added and delete the existing column name.
3. Type in a single quote ('), followed by the hard-coded value, followed by another single quote (as in 'smith@KnowledgeSync.com').
4. Go to the "Customized Name" for this column and enter an appropriate name for this field.

3.6.3.22 Creating a Column Pointing to a File Spec

One of KnowledgeSync's special functions is to create a column whose contents are a file spec – that is, the column refers to the name and location of one or more files that you wish KnowledgeSync to deliver. Multiple files (referenced within the column's value) must be separated by commas.

The file spec referred to within this column must use either the actual server, disk and directory spec, or must use UNC format; do not refer to mapped drives.

When you add such a column to a query, the Customized Name of that column **must be** "attachment_files" (case sensitive).

When an event uses a query that contains such a column, you'll have the option to instruct KnowledgeSync to deliver the corresponding file(s) via email, webcast, or Copy/FTP.

In a web service call or VB script, if a response parameter was named "attachment_files", KnowledgeSync will use that as the name of the file to be attached.

3.6.3.23 Creating a Column Pointing to a Dynamically-Named File Spec

You can modify the use of the above "attachment_files" option to specify a dynamically-named file by referring to a column name (e.g., "{order_no}") within the associated file's name.

For example, you might have multiple files that store customer order tracking details in your 'C' drive. These files are named according to the order they relate to as in:

Tracking_100.pdf
Tracking_101.pdf
Tracking_102.pdf
(and so on)

(The above 3 files contain tracking details for orders #100, #101, and #102.)

As long as your query retrieves the "order number" value (e.g., in a column named "orders.order_number"), you can have KnowledgeSync to pick up the files associated only with the order records that meet your query's criteria. In this case, you would modify the name of the selected column to read:

```
'C:\Tracking_{order_number}.pdf'
```

You refer to the customized name of the desired column, within curly braces. KnowledgeSync will automatically substitute in the appropriate order numbers and retrieve the corresponding PDF files.

Make sure that the customized name for the column that retrieves these files is "attachment_files" (do not include the quotation marks) and be sure that the options to "include files", include "reports", and "reports must wait" are ALL checked 'on' in any events that use this query.

3.6.3.24 "If – then – else" Logic

Consider the following logic:

"if the contact's email address is blank, use the default account email address"
"if the contact's email address is blank, use the salesrep's email address"
"if the value of the credit status field is "H", I want the alert message to read "Hold""

To check the value of a field and then modify that value (or use a different value) is very useful. The SQL syntax for doing so is:

```
CASE table.column when 'value' then table.column+'value' else table.column END
```

If, for example, you have a column in a table that has values of I, C or D and you want to display them as IN, CM, and DM in your alert message, you'd create three columns in your query as follows:

```
CASE table.column when 'I' then table.column+'N' else table.column END
CASE table.column when 'C' then table.column+'M' else table.column END
CASE table.column when 'D' then table.column+'M' else table.column END
```

The preceding syntax is for SQL Server databases.

3.6.3.25   Removing Columns from a Query
To remove a column from a query, go to the "Selected Columns" window, click on the red Minus sign to the left of the field you wish to remove.

3.6.3.26   Return Distinct Rows of Data Only
What if you wished to receive a list of all customers who placed orders last month – but didn't wish to see any duplicate records (such as if a client purchased multiple times during the month)? This is referred to as "returning distinct records only" (eliminating any duplicate records) and is available via a checkbox in the "Selected Columns" window.

"Return distinct" is based on the columns chosen for a query, so (using the previous example of client orders), if you choose just the client's name, your results will show one record per client. If, however, you choose (as columns) both the client name and their order number, a client who purchased multiple times during a month would appear (in the query's result set) multiple times, as each order number is "distinct".

3.6.3.27   Sub-Queries (Embedded Selects)
One of the more advanced query design functions is the ability to create a column whose value is the result of another query. This is referred to as "sub-queries" or "embedded selects".

Embedded selects require advanced SQL syntax knowledge, as each column name would be edited to include a SQL select statement.

> Embedded selects must use the same database provider as the query in which they are embedded.

To create, view, or modify embedded selects, use the Calculated Column Editor, as it gives you a full-screen editor in which to create your embedded select statement.

KnowledgeSync has an EventPak that illustrates the use of embedded selects. To receive this Pak, please contact Customer Support using the form on our website.

3.6.4   Sort Type (Record Sorting Order)
When a query runs, it retrieves matching data records. You can specify the order in which these records will be listed and processed. (And this is the order in which those records will be shown in any corresponding alert messages.) This is the query's "sort type."

Any fields that were selected for the query's "Columns" are available for use in its sorting order. To select your sorting order, go to the "Selected Columns" window, locate your first field for sorting purposes (you can have multiple levels of sorting), click in the "Sort Type" (it will default to "None"), and select either "Ascending" or "Descending".

The selected column will now appear in the "Sort Order" window.

If you choose multiple columns for sorting, you can re-order those selected columns using the up and down arrows in the "Sort Order" window. (And you can use the 'x' button to remove columns previously selected for sorting.)

The "Sort Order" window will appear like the following:
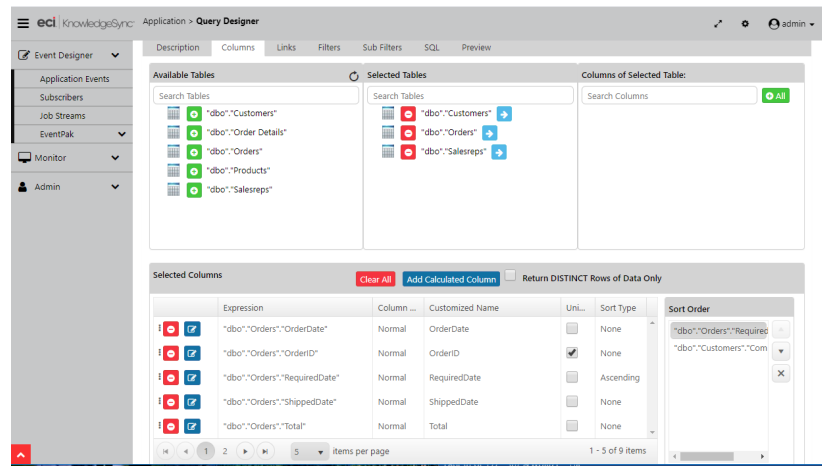
QUERY "SORTING" (BOTTOM-RIGHT)

3.6.4.1 The Effect of Sorting on "Grouped" Alert Messages

"Sorting" controls the order in which matching records are displayed in alert messages. This is especially important when you want an event to "group" records together – such as sending each employee an alert that lists all of their overdue activities or sending a customer an alert listing just their orders.

In order to send alerts where similar records are grouped together (i.e., all of an employee's activities, all of a customer's orders) you must use Sorting to group those records accordingly within the query's result set.

(In the above examples, if you do not sort the records, an employee or salesrep might receive multiple alerts, each alert containing one, or just a few of their records, instead of a single alert containing all of their records.)

In the preceding examples, the "overdue activities" query would sort by rep, the "customer order listing" query would sort by customer number (or name).

3.6.5 Links

If you have chosen more than one table, you must tell KnowledgeSync how those tables are linked (or "joined") together. It is essential that you choose the query's primary table first in your series of links.

In the example of a "Jobs; Created Today" query that uses the "jobs" and "organizations" tables, this query's primary database table is the "jobs" table.

Link your tables by doing the following:

Click on the "Links" tab. You will see the following:
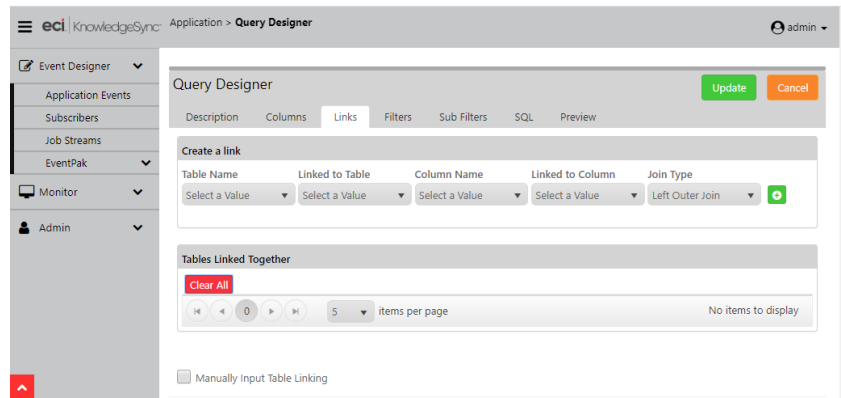
QUERY "LINKS"

To add a new link:

- o Under "Table Name," select the query's <u>primary database table</u>.
- o Under "Linked to Table Name," select (or confirm) a corresponding table.
- o Under "Column Name," select the "link from" field.
- o Select the "link to" column name. If a column in the "Link to Table" is found with the same name as the "Link from Table," that column will be loaded into the "Linked to Column Name".
- o The link type will default to "left outer join"; leave this unless your database schema specifically requires a different type of join.
- o Click on the green Plus sign to add the link.

Once one or more links are added to a query, this tab will look like the following:
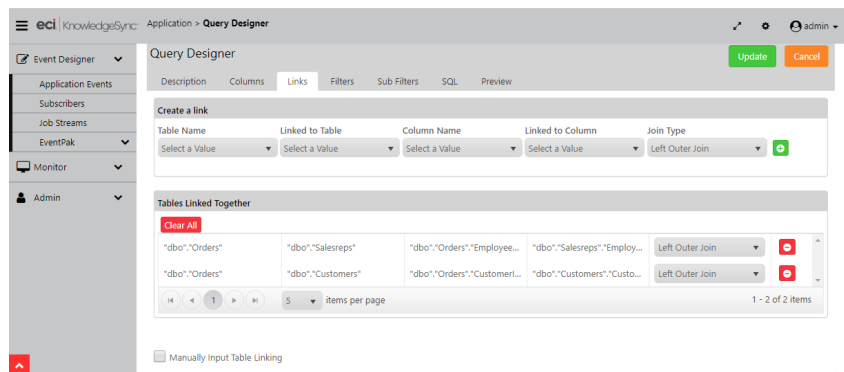
SUCCESSFULLY ADDED LINKS

If you have more than two tables that require linking, repeat the above process until all tables are linked. In most cases you will have a number of links that is equal to one fewer than the number of tables the query is using.

3.6.5.1 Manually Input Table Linking

Depending on the applications, databases, and connection method (.NET Provider, OLE/DB, ODBC) you are using, you may come across an instance where KnowledgeSync's default linking syntax is not sufficient for the type of "join" you wish to create.

To specify your own table linking syntax, follow these steps:

1. Join the two tables together using the table linking wizard.
2. Click on the SQL tab and copy the "join" syntax (starting <u>after</u> the word "from" and ending before the ')' symbol. Do not include the parentheses that begin and end the join syntax. E.g.,

```
dbo.Jobs LEFT OUTER JOIN dbo.Organizations ON dbo.Jobs.jmpCustomerOrganizationID =
dbo.Organizations.cmoOrganizationID
```

3. Click back on the Links tab, click on the checkbox titled "I would like to manually input the table linking." A blank line will appear.
4. Paste the "join" syntax that you copied from the SQL tab into this line.
5. Edit this syntax to perform the join as you require.

You may also directly edit the field names in the "Tables Linked Together" grid, detailed in the following section.

If you manually edit the linking syntax for a query, that query is no longer eligible for KnowledgeSync Support Services. (Thus it is highly recommended that you use the technique detailed in the <u>following</u> section for modifying a query's links, as this method will keep the query eligible for technical support services.)

3.6.5.2 *Modifying Your Links*
Do <u>not</u> select "I would like to manually input the table linking".

Under "Tables Linked Together", locate the link you wish to modify.
Click your cursor in any of the "column name", "linked to column", or "join type" fields.
Manually type in (or select) the desired syntax. Any valid SQL syntax is supported.

3.6.5.3 *Multi-Column Links*
To link one table to another using <u>more than one column</u>, follow these steps.

Create a link using the first of the two fields you need to link with.
Click in the "Linked to Column" field; an edit-capable window appears.
Add a blank space after the column name.
Type in "and table1.2nd_link_column = table2.2nd_link_column"
In place of "table1.2nd_link_column", key in the secondary linking column in the first table, and in place of "table2.2nd_link_column", key in the secondary linking column in the second table. Do not include the quotation marks in your syntax.
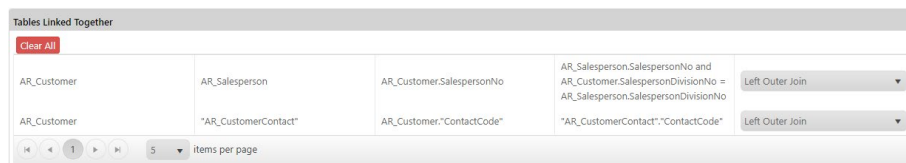
A MULTI-COLUMN LINK

3.6.6 Filters
The most important part of query design is the "conditions" that will cause the query to trigger an event. These conditions are referred to as "filters" and "sub-filters".

There are four general concepts of filter design. They are:

- What column values do you need to test
- What "operator" will a filter use
- What will the column value be compared against
- Should a filter be parameter driven

The following sections address each of these concepts.

3.6.6.1   Available Column Values

Filters can check the value of any column within any tables or views within a database. They can identify new records, records that meet specific criteria or thresholds, and can also check to see if the value of a field has changed.

3.6.6.2   Operators

A filter can use the following operators (along with "is" or "is not"):

- Equal to (priority is equal to "high")
- Between (use "greater than or equal to" and "less than or equal to" instead)
- Greater than (amount is greater than 100)
- Greater than or equal to
- Less than (amount is less than 100)
- Less than or equal to
- Like (contains the string of letters "brit")
- Null
- One of (priority is one of "Critical", "High", or "Medium")
- Starting with (starts with the letter "brit")

A filter can also use one other operator:

Value has **changed** since the last time KnowledgeSync checked it

3.6.6.3   Compare Values

A filter can do much more than compare the value in one field against a constant value (e.g., test for a priority of 'high'). Here are some examples of other tests that can be executed by a filter:

Test a field against another field
    (*sale price is less than the cost of goods*)

Test a calculated field against a constant value
    (*Sales amount * sales probability is greater than 10,000*)

Test one calculated field against another calculated field
    (*percentage of damaged items > percentage of undamaged items*)

Test a numeric field against a range
    (*sales probability is greater than 60% and less than 90%*)

Test a date field against a date variable
    (*forecast close date is equal to "today"*)

Test a calculated date value against a threshold
    (*due date is within 7 days*)

Test a calculated date field against a variable
    (*expire date is this month*)

3.6.6.4   Parameter-Driven Filters

Consider this scenario:

Your sales manager wishes to be alerted about pending sales greater than $25k. Your CFO wants to know about sales greater than $50k. And your CEO wants to know about sales greater than $100k.

Do you need three different queries? The answer is <u>no.</u>

You can design one query called "<u>Sales > 'x' Dollars</u>.

This is a "parameter-driven" filter. Three different events can each use this same query but specify a different triggering threshold (sales amount) on the event level.

3.6.6.5   Filter Components
When you click on the "Filters" tab you will see the following:
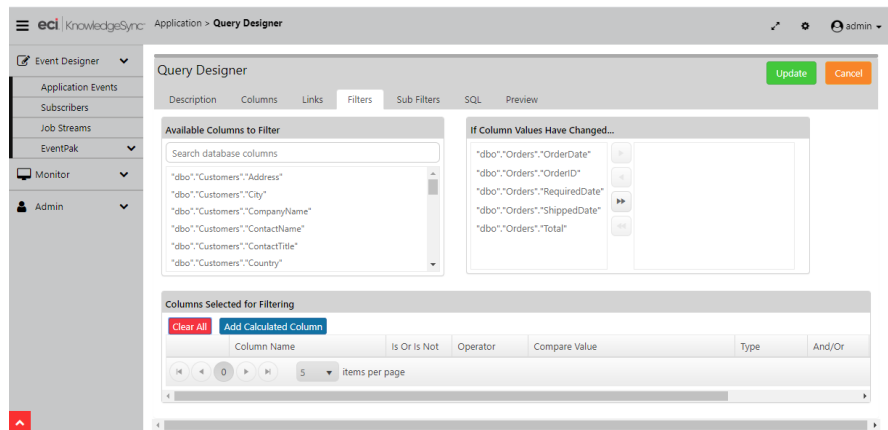
QUERY "FILTERS"

When creating a filter, you can select a specific column (from the list of "available columns to filter") whose value you wish to test, and/or you can select one or more columns (from the list of "if column values have changed") should you wish to monitor a column for a change to its value. (See "Value Change Filters").
To select an available column to filter, double-click on the desired column. To create a filter based on a calculated column, click on the "Add Calculated Column" button. To edit a selected column (such as to add a calculation to it), click on the blue "Edit" button; to remove a previously selected column, click on the red Minus sign.

Note the following:

Column Name. The name of the field you wish to filter on.
"Is" or "Is Not". Defaults to "is". Click next to this word and choose "is not" if applicable.
Operator. Defaults to "equal to". Click next to this word and choose any other operator if applicable. See "Operator Notes" for details.
Compare Value. The value to compare the selected column's value against. This may contain a constant value, a value from another database field, a calculation, a date substitution, or a question mark. (See following section.)
Type. The type of data the selected column contains. Leave with its default unless doing a date calculation (use 'number') or comparing against another database field or calculated value (use 'literal').
"And / Or". The ability to specify "Boolean" logic, e.g., ("look for condition 'a' and/or condition 'b'"). The "and/or" logic is <u>sequential</u>; this means that a filter's "and/or" applies to the current filter _in relation to the next filter_. Parenthetical or "grouped" and/or logic is supported. See "And/Or Logic".
Optional Prompt. The optional prompt is used only with parameter driven filters; see "Parameter-Driven Filters" for details.

3.6.6.6   Comparing a Column against a Constant Value
Example: "ship via = FedEx"
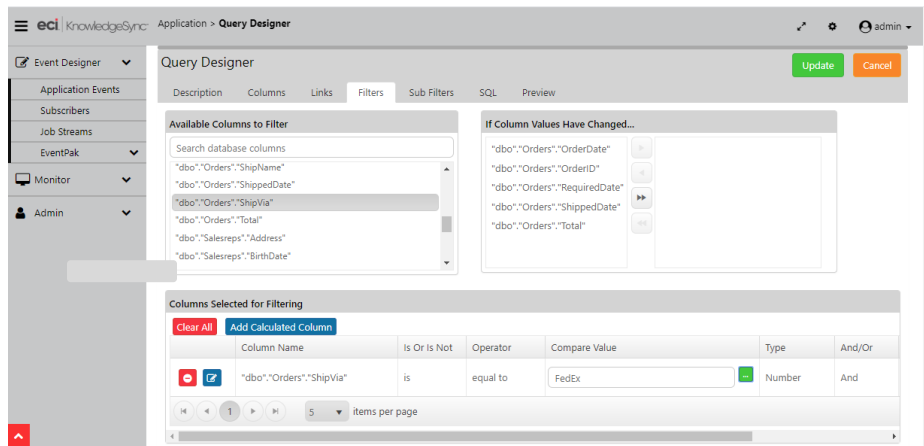
COMPARING A COLUMN VALUE TO A CONSTANT VALUE

You can manually enter the compare value, or you can choose it from a drop-down list. KnowledgeSync <u>is</u> case-sensitive. To select the value from a list, follow these steps:

Click on the green "..." button in the "Compare Value"field.
Click on the list button under the heading "Type in a Value or Choose from dropdown List". This will display a list of possible entries for this field.
Choose the value that you wish to compare the column to and click on "OK".

> *The dropdown list shows values that <u>have appeared</u> in this field in the underlying database. It does not display all possible values that <u>could appear</u> there.*

3.6.6.7   Comparing a Column against another Database Field
Example: "shipped date is later than the required date"
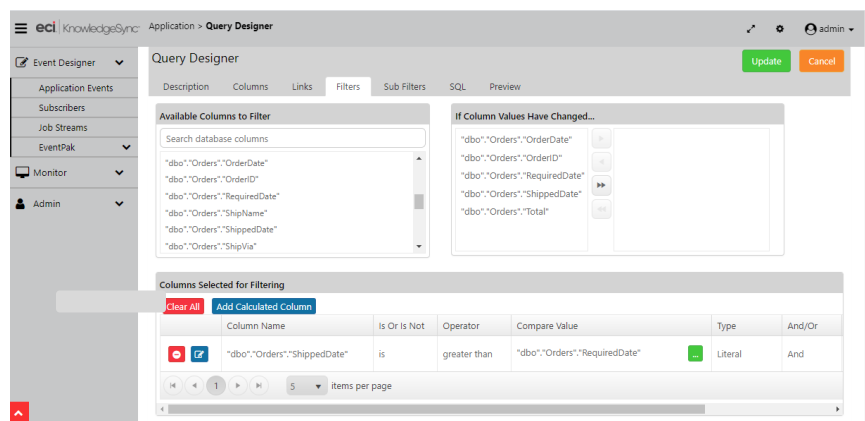
**COMPARING ONE COLUM'S VALUE AGAINST
ANOTHER COLUMN'S VALUE**

To do this:

Click in the "Compare Value" field and delete the '?' that appears there.
Type in the full name ("table.column") of the field you wish to compare the selected column to (e.g., "dbo"."Orders"."RequiredDate").

Go to the "Type" field and select "Literal" (so that KnowledgeSync recognizes the 'compare value' as a database column name).

3.6.6.8 Comparing a Column against a Date or Numeric Range
Example: "order total is greater than $99 and less than $999"
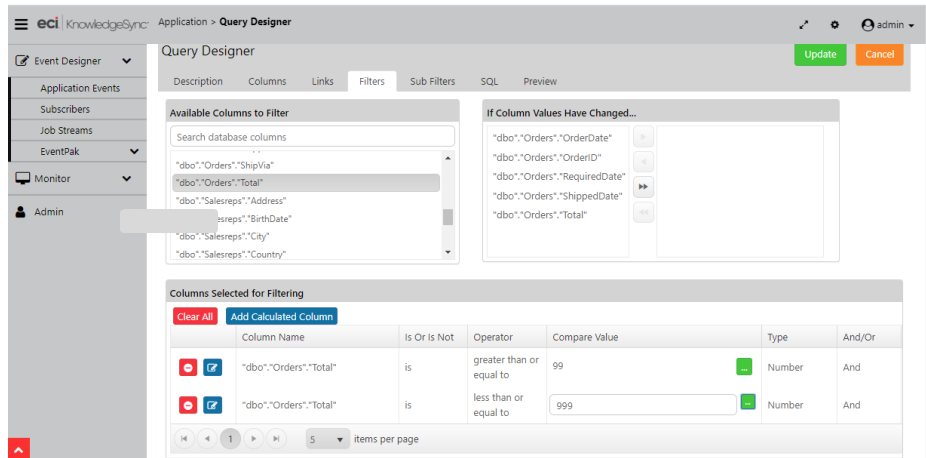
COMPARING A FIELD AGAINST A DATE OR NUMERIC RANGE

A common request is to see if records fall within a date or numeric range. Although you can use the "between" function, use of this operator is **not suggested**, as this operator tends to execute less efficiently than using a combination of "greater than or equal to" and "less than or equal to". Additionally, not all database query methods support "between", and this operator cannot be used in parameter-driven filters.

To create a range filter, follow these steps:

Select (as filter #1) the field that you wish to perform a "range" test on.
Choose the operator "greater than or equal to"
In "Compare Value", remove the '?' symbol and key in a "beginning" range (such as a "start date" or lowest allowable threshold).
Make sure the "And/Or" option reads "And".
Add a second filter on the same field as filter #1.
Choose the operator "less than or equal to"
In "Compare Value", remove the '?', key in an "ending" range (such as an "end date" or highest allowable threshold).

3.6.6.9 Comparing a Column against a Date Substitution Variable
Example: "order date = today"
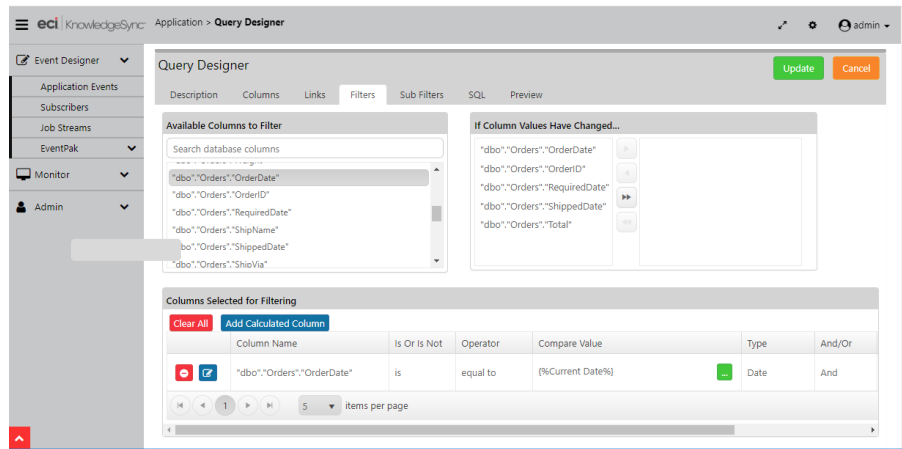
COMPARING A FIELD AGAINST A
DATE SUBSTITUTION VARIABLE

Many queries test the value of a date field against a value such as "today", "tomorrow", "last week", or "next month". KnowledgeSync provides a wide variety of "date substitution variables" that can be used as the "Compare Value".

To use a date substitution variable as a compare value, follow these steps:

Click on the "…" button in the "Compare Value" field. In the resulting window, look for the list of "Variables to use in calculation".
There is a wide variety of variables to choose from. You will see multiple formats for the same variable. The choices without any formatting suffix will use your system's regional settings for its default date format.
Double-click on the value that you wish to compare the column to and click on "OK".

Be careful when using the "current date" variable if you are comparing it against a field that contains both the date and time. A field that contains today's date and time will always be greater than (and not "equal to") the "current date" variable.

You can address this either by configuring your filter see if the date/time field is greater than or equal to today's date, or by configuring your filter to see if the date/time field is greater than "{%Current Date%} 00:01" and less than "{%Current Date%} 23:59".

3.6.6.10 Identifying "New" Records
You can configure a filter to retrieve records with a create date of "today", and since KnowledgeSync remembers what records have already been triggered, it will not notify you repeatedly about the same records.

But KnowledgeSync has a special date variable that makes this process more efficient – it's a variable called "**Last Date/Time Event Checked**". This variable allows a query to retrieve records that have been added to your database only since the last time KnowledgeSync ran the event.

This is particularly useful if your database is quite large or you are running an event very frequently (e.g., every hour or less).

This variable will be formatted according to the KnowledgeSync server's regional settings for date format.

3.6.6.11 Pervasive & Providex Date Formatting
PSQL and Providex databases typically store their dates in "YYYY-MM-DD" format. Thus to compare a date value from one of these databases to a date variable, you would typically use a variable which shows the matching format, such as:

```
{%Current Date YYYY-MM-DD%}
```

3.6.6.12   Parameter-Driven Filters

Requirement: "3 events; for invoices coming due in 7, 14, and 30 days"

Sometimes it's more efficient to create queries whose triggering conditions are specified on the <u>event</u> level as opposed to (or in combination with) a <u>query's</u> filters. For example:

> Invoices due for payment in the next 7, 14, and 30 days

In this scenario, you could create <u>3 queries</u> (one query for each triggering threshold), or you could create <u>1 query</u> – where the "number of days" is a threshold ("parameter") specified on the event level. A "parameter-driven" query enables you to use the same query in multiple events, each event having a unique triggering parameter (such as '7', '14', or '30' days).

A parameter-driven filter is represented by the '?' symbol in the Compare Value field of a query – the '?' indicating that the compare value will be specified on the event level.

Events using parameter-driven filters will require that a compare value be specified at the time that the event is <u>configured</u>. Once the event is configured, the parameter will have a specific value which is used every time that the event is executed.

The operators "one of", "between", and "null" may <u>not</u> be used as parameter-driven filters.

To set up a filter to be parameter-driven, follow these steps:

Select the column to be filtered and its operator.
Make sure that a '?' appears in the "Compare Value" field.
In the "Columns selected for filtering" window, scroll the window to the right until you see the field called "Optional Prompt." The optional prompt is the text that will appear (on the event level) when this query is selected for use with an event.   (The "optional prompt" appears within an event using a parameter-driven query filter. The "prompt" refers to the text that will inform a person that a triggering value is required for this filter, such as "*How many days until an invoice is due?*")

> *The value for a parameter-driven filter must be specified <u>when the event is configured</u>, as KnowledgeSync will <u>not</u> prompt for any values when the event is scheduled to run.*

For a parameter-driven filter, delete the default optional prompt text and key in your own text, such as:
  - o   "Select a Priority:"
  - o   "Status is:"
  - o   "Date Range Begins" or "Date Range Ends". (This requires two parameter-driven filters on the same 'date' field.) Your filters will look like this:
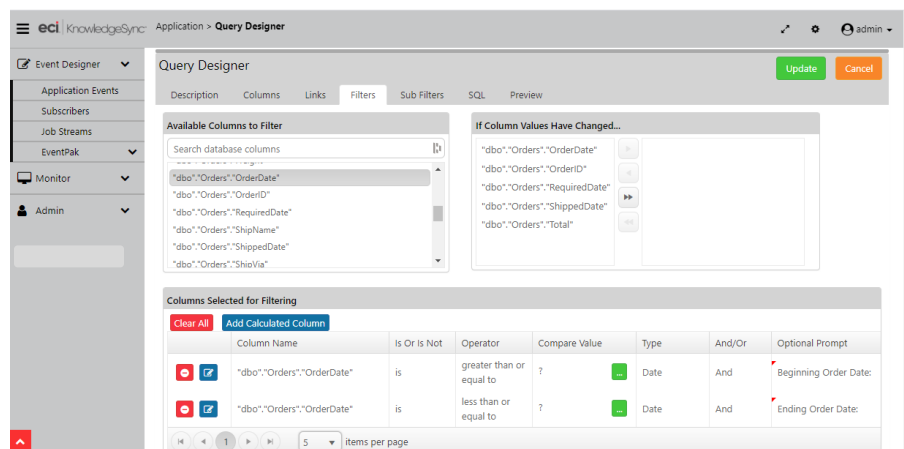
MAKING A FILTER "PARAMETER-DRIVEN"

3.6.6.13 Checking a Column Against a "Changed" Value

Example: "credit status has changed"

KnowledgeSync can tell you when the value in a data field (column) has changed. When a "data change" event runs for the <u>first</u> time, KnowledgeSync stores the value of the field(s) being monitored for a change.

The next time (and every subsequent time) the event runs, it compares the "current value" of the field(s) being monitored to the "old value" from the previous run. If the two values are different, the event is triggered and KnowledgeSync replaces the "old value" with the "new current value". And the cycle begins again.

Note the following about "change-based" queries:

They may check for changes to fields in the query's <u>primary</u> table only.
They <u>must</u> have a unique column, this column cannot be a calculated or edited column, and its "Customized Name" must <u>not</u> be changed from its default value.
If the database you're monitoring supports "auditing", it's more efficient to build queries to monitor that database's audit table rather than using changed-based queries.
If you have multiple events checking for a change <u>in the same field</u>, these events <u>must</u> be scheduled to run at the same time (e.g., daily at 9:00 AM).
You can check to see if a field value <u>has changed</u>, or if it <u>has changed to a specific value</u>.
If a query monitors multiple "change" fields, KnowledgeSync assumes an "OR" relationship between them.
If a query includes "change-based" criteria and standard filter criteria, KnowledgeSync assumes an "AND" relationship between them.
Any events using a change-based query <u>must</u> have "Repeat Notification for Triggered Items") checked <u>on</u>.
When you Preview a query, the "change" logic is ignored.
Change-based queries are possible only for tables that have a single-column unique. If a table uses a multi-column unique, please create a view for that table that has a single-column unique value.

3.6.6.14 Selecting a Column for Value-Change Tracking

To create a filter that checks to see if a value has changed, go to the Filters window titled "**If Column Values Have Changed**" and you'll see the columns from the query's primary database table. To select a column for change-tracking, select the column from the left and click on the right-arrow button to move it to the right.

You may add additional filters, including a filter on the <u>current</u> value of your "change field" to check for records whose content has changed <u>to</u> a specific value.

3.6.6.15 Storing & Reporting On "Changed" Values

To store the history of when a "changed value" event has been triggered, it is suggested that you use the "Create File" action to create an audit trail. This lets you store the details of what records were changed, when they were changed, and what the old and new values for the change are.

3.6.6.16 Filter Operator: "One Of"

The "one of" operator lets you test a column for <u>any one of multiple values</u> such as:

*Send an alert for any calls that have a priority of 'A' **or** 'B, **or** 'C'.*

To use the "one of" operator:

Select "one of" from the operator list.
Click in the "Compare Value" field and remove the '?'.
Click on the "…" button in the "Compare Value" field.
In the "Type in a Value" field, either key in the individual compare values (pressing 'Enter' after each entry), or choose the compare values from the drop-down list.
Click on "OK" when complete

The "one of" operator may <u>not</u> be parameter-driven.

3.6.6.17 Filter Operator: "Between"

The "between" operator is a slow-executing one, and as such we suggest that you use the combination of the "greater than or equal to" and "less than or equal to" operators in place of the "between" operator wherever possible.

The "between" operator may <u>not</u> be parameter-driven.

3.6.6.18 Operator: "Starting With"

The "starting with" operator lets you locate any database records that have a field whose value <u>starts with</u> a certain string of letters, numbers, and/or symbols. To use the "starting with" operator, follow these steps:

Select "starting with" from the operator list.
Edit the "Compare Value" field and type in the first few characters of the string that you wish to use for matching purposes.

The "starting with" operator <u>may</u> be parameter-driven.

3.6.6.19 Operator: "Like"

"Like" is similar to "starting with" except that it lets you locate records that have a field whose value <u>contains</u> a certain string of letters, numbers, and/or symbols.

 "Starting with '<u>brit</u>'" will retrieve "Britain" and "Britannia" but not "New Britain".

"Like '<u>brit</u>'" will retrieve all three values.

To use the "like" operator, follow these steps:

Select "like" from the operator list.
In the "Compare Value" field, enclose the text you wish to monitor for within percent signs, as in "%brit%".

The "like" operator may be parameter-driven.

3.6.6.20 Operator: "Null"

You might wish to check to see if a field's value is blank; but a "blank" value can be either <u>blank</u> or <u>null</u>. To test a field for a blank value, leave the compare value field blank. To test a field for a null value, follow these steps:

1. Select "null" from the operator list.
2. Remove any value in the compare value field.

If you are not sure how an application identifies an "empty" field, you might want to create a query with <u>two</u> filters – one that tests for blank, and one that tests for "null". Be sure to use the "or" condition between them.

3.6.6.21 Using Calculated Fields in Filters

Please refer to the earlier section of this chapter titled "Calculated Fields" – and be sure that the query filter's "Type" field ("data type") refers to the correct type of data being returned by your calculated field.

*Calculated fields can be used in **both** a filter's <u>column name</u> field and in the <u>compare value</u> field. Calculated field filters may be parameter-driven.*

3.6.6.22 And/Or ("Boolean") Logic

Consider the following scenario:

Notify me about new orders for greater than $99 <u>and</u> . . .

. . . that have freight charges less than $5,

<u>or</u>:

. . . that have freight charges more than $50

This type of "and/or" ("boolean") logic requires the ability to group filters together, using parenthetical expressions. Each filter that you create automatically has an "and/or" associated with it – and those words apply to the current filter AND the filter that follows it. If there is no following filter, the "and/or" has no effect.

So – to create the preceding scenario:

Create the three filters the query needs:

1. Orders greater than $99; ("and")
2. Freight charges less than $5; ("or")
3. Freight charges more than $50; ("and/or" does not apply)

Edit the "Column Name" of filter #2 and place a left parentheses in front of the name.
Edit the "Compare Value of filter #3 and place a right parentheses after the compare value.

> *If the compare value is a <u>character</u> or <u>date</u>, place single quotation marks around the compare value, and change the "Data Type" to "Literal.".*

Your Boolean filters will look like the following:
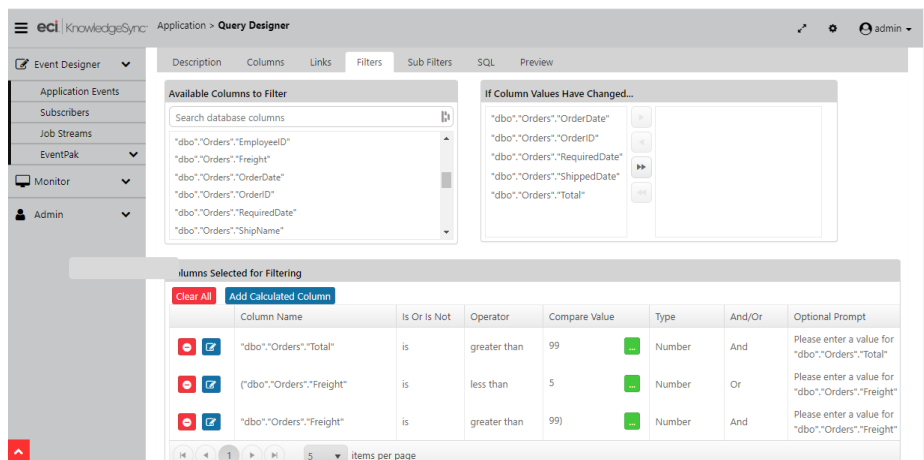
BOOLEAN ("AND/OR") LOGIN IN FILTERS

3.6.7 Sub-Filters

Sub-filters are not used with record-level or data-change queries; they are used only with aggregate queries and will be discussed in this chapter's section on aggregate queries.

3.6.8 Query SQL

> *KnowledgeSync Technical Support <u>does not</u> support manually-edited SQL.*

The "SQL" tab lets you review the SQL syntax that a query is executing. There is rarely (if ever) the need to make direct changes to a query's SQL. Once you start manually editing query SQL, all subsequent changes to the query <u>must</u> be done through the SQL tab.

If you wish to manually edit the SQL, place a checkmark in "I would like to manually edit the SQL for this query" box. It is essential that you select all needed columns for a query and identify the query's unique column <u>before</u> you manually edit a query's SQL.
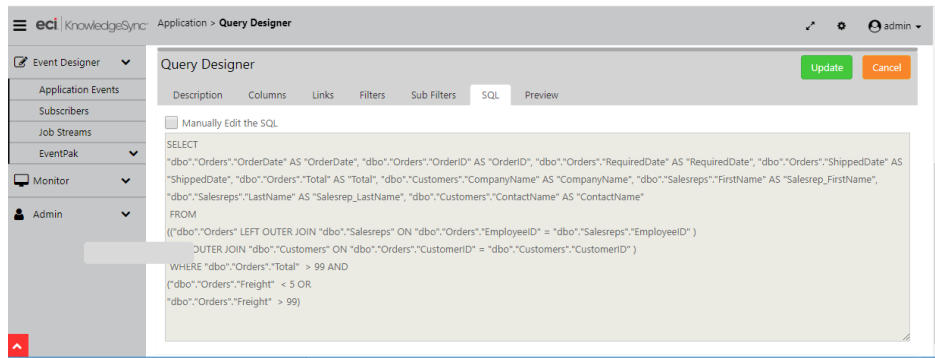
QUERY "SQL" SYNTAX

Once you manually edit the SQL syntax, any modifications that you make to any of the other query definitions tabs will <u>not</u> be retained.

If you edit a query's SQL and then de-select the manual editing option, the query will re-constitute the SQL syntax based on the content from the other query tabs.

3.6.8.1   <u>*Using a Stored Procedure in a Query*</u>
To use a stored procedure instead of a query, follow these steps:

Select the option "I would like to manually edit the SQL for this query" (see preceding section).
Type in the procedure name, as in "dbo.myprocedurename"
To pass parameters into your procedure, use the following syntax:
"dbo.myprocedurename param1,param2,param3"
(Parameters that are string values must be enclosed by single quotes.)

As stored procedures are custom elements in KnowledgeSync, our Technical Support Services are not able to assist you in debugging these procedures.

3.6.9   Preview
The Preview tab lets you verify that your query is running as you wish it to.

> *Don't wait until you have built every aspect of a query before previewing it. Use the preview function repeatedly as you add more and more elements to a query and begin to refine its criteria.*

When you click on "Preview", KnowledgeSync will return any records that meet your query's criteria. If the preview tab is blank (containing no records), it means that no records met your criteria. When Preview mode returns matching records, it will appear similar to the following:
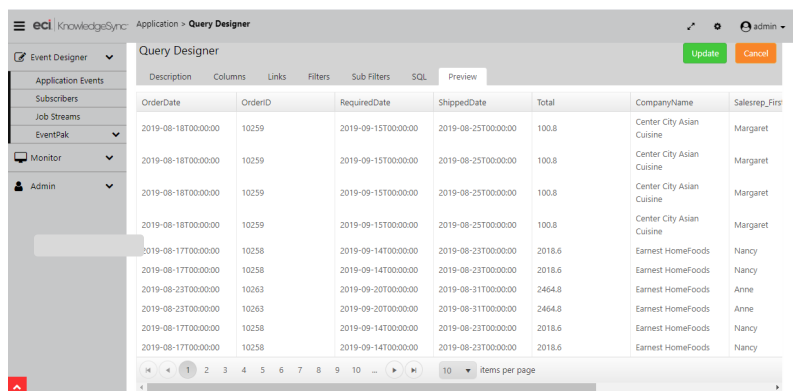
QUERY "PREVIEW"

If you receive an error message when previewing, it means that KnowledgeSync was not able to successfully execute your query. The most common reasons for this are:

- The database connection (e.g., .NET Provider, ODBC, OLE/DB) for this application was not correctly configured
- Multiple tables for this query were not correctly linked
- Your query includes calculated fields that contain invalid SQL syntax
- You have specified an invalid customized name for a query column
- You have specified an invalid filter
- You have edited the underlying SQL and introduced a syntax error
- You have referenced a SQL function not supported by your database connection

If your query includes parameter-driven filters, you will be prompted to enter corresponding values; if the filter requires date values, you will be able to choose from KnowledgeSync's date substitution variables.

3.7 Saving a Query

To save your query, click on the "Save" or "Update" buttons.

3.8 Copying a Query

To make a copy of a query, follow these steps:

Highlight (single-click on) the query you wish to copy.
Click on "Copy Query."
Click on "Paste Query."
Go into either of the two (identical) queries and modify it as needed – and don't forget to change the query's description!

3.9 Deleting a Query

Before deleting a query, make sure that no events use it. If you delete a query that is linked to an event, the event will fail to execute successfully. To delete a query:

Under the list of "Query Definitions", locate the query you wish to delete and scroll the window as far to the right as possible.
Click on the red "Minus" button at the far right.

There is no "Undo" option once you have deleted a query.

3.10 Locating a Specific Query

To locate a specific query, you can either use the page numbers at the bottom of the "Query Definitions" window, or you can use the "Search Description" field to type in all or part of the name of the query you're looking for and (if it exists), you'll be brought to the desired query.

3.11 Aggregate Queries

Aggregate queries are queries that are triggered by a group of records that collectively meet a certain condition. For example:

More than 10 open service calls with a priority of "critical"
Fewer than 5 forecast sales due to close next week
Clients who have purchased more than $50,000 of a specific product.

Aggregate queries do not retrieve "details" about individual records; instead, aggregates queries retrieve data pertaining to the aggregate condition being met. For example:

Condition to Monitor for: More than 10 open service calls
Condition that Exists: 14 open service calls

Query Retrieves 1 Record: "14"

Condition to Monitor for: Clients with > $1,000 in sales last month
Condition that Exists: 3 matching clients
Query Retrieves 3 Records: "Smith 2500", "Jones 1300", "Davis 3100"

Aggregate queries cannot use "change-based" filtering.

3.11.1 Aggregate "Grouping" Results

One of the benefits of aggregate queries is that they can "group" their results based on any fields in a database. Consider the following three examples and the results from corresponding aggregate queries:

KnowledgeSync can "count" the number of records that meet certain criteria. For example, an aggregate query that counts the number of orders today could yield the following result:

```
17
```

An aggregate query that counts the number of orders per region today could yield the following result:

```
6       Northwest
3       Southwest
8       Eastern
```

And an aggregate query that counts the number of orders per salesrep within each region today could yield the following:

```
3       Northwest      Karen Chavez
2       Northwest      Dave Bruce
1       Northwest      Sue Murphy
```

3.11.2 Aggregate Query Configuration

Please read through the earlier (record-level) query design section of this chapter before proceeding with the following sections. The instructions on aggregate query configuration assume that you are familiar with record-level query design and so the following sections will focus only on those areas of aggregate query design that are different from record-level queries.

3.11.3 Aggregate Query Columns

An aggregate query's columns are selected based on the following three criteria:

The field that the aggregate function will be performed on
The "group by" field(s)
Other fields to appear in the alert message, contain a recipient's address, or will be used in an event's actions

3.11.3.1 Aggregate Function: Summarize

The "summarize" function lets you total the amounts in a numeric field, for example:

The sum total of dollars of sales received today
The sum total of minutes spent on high priority support calls
The sum total of units of a particular item sold this month

3.11.3.2 Aggregate Function: Average

The "average" function lets you derive the numeric average for a field, such as:

The average sale price this week
The average length of a support call in minutes

The average percent of damaged received goods

3.11.3.3 Aggregate Function: Count
The "count" function lets you <u>count the number of records</u> that meet certain conditions, such as:

The count of overdue invoices per client
The count of activities scheduled per rep today
The count of current backorders for a specific stock item

3.11.3.4 Aggregate Functions: Maximum & Minimum
The "maximum" and "minimum" functions let you identify <u>the maximum or minimum values for a field</u>. This function is for such things as:

The maximum sale closed last week
The minimum order quantity this week
The most recent contact date for a client

3.11.3.5 Aggregate Grouping
When working with aggregate queries, you need to consider not only what you want to perform the aggregate upon (e.g., total <u>sales</u>) but whether you wish to "group" the results of that aggregate (e.g., sales total by salesrep).

The ability to create aggregates that group their results is one of the most powerful aspects of aggregate queries. Consider how you can take an ordinary aggregate query such as "total sales" and create the following by adding one or more "group by" fields:

Total sales by salesrep
Total sales by region
Total sales by product line
Total sales by region and (within region) by product line
Total sales by regional sales manager

All aggregate functions support grouping, but grouping is most commonly used in "summarize", "average", and "count" aggregate queries.

3.11.3.6 How to Configure an Aggregate Query
To select columns for a "summarize", "average", or "min/max" query, follow these steps:

Identify the <u>aggregate condition</u> you wish to perform (e.g., you wish to "<u>total</u>" the <u>sales amounts</u>).
Select the <u>table</u> that contains the field you wish to perform the aggregate function on (e.g., the sales amount is stored in the "<u>orders</u>" table).
Select the <u>column</u> you wish to perform the aggregate function on (e.g., the "total" column from the "orders" table). The aggregate column is usually the first column selected in an aggregate query; the exception to this is aggregate queries that retrieve data for charts. Those queries need to have their "group by" value selected first, and the aggregate value selected second.
Under "Selected Columns", click next to the word "Normal" in the Column Type field to display the list of aggregate functions and select the appropriate function (e.g., "Summarize").
Change the customized name of this column to reflect that this field contains an aggregate total (e.g., "sales_total").
(Do <u>not</u> put the aggregate function name at the <u>start</u> of a customized name (e.g., "total_sales") as some databases do not allow this.)
Go back to the list of available columns and select your "group by" field(s) (e.g., select the "customer" column to total the sales by customer).

Your query will look similar to the following:
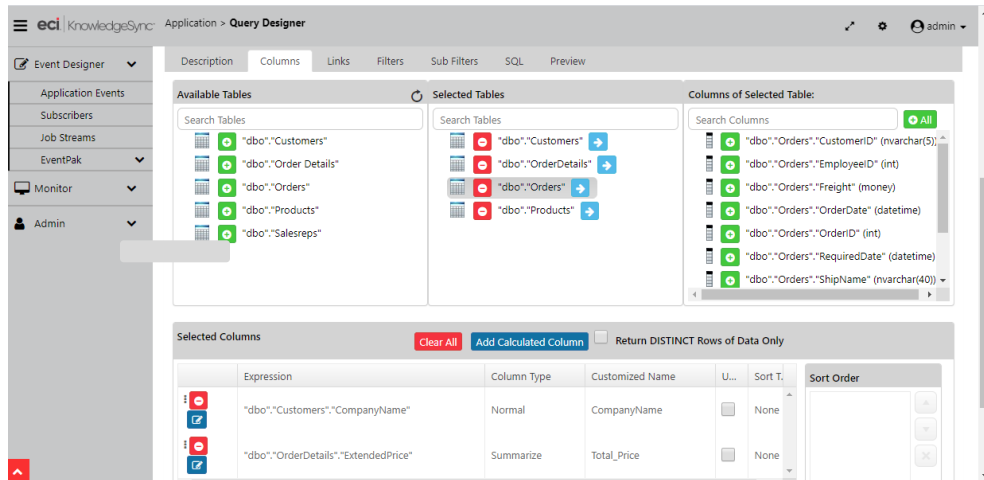
AGGREGATE "SUMMARIZE" FUNCTION:
TOTAL SALES PER CUSTOMER

3.11.3.7 The "Count" Function

To create an aggregate "count" query, do the following:

On the Columns tab, choose the column that's the primary key (or "unique") for the table whose records you wish to count
Click next to the word "Normal" in the Column Type field and select "Count."
Change the "Customized Name" as appropriate
Select your "group by" field(s)

Your result will look similar to the following:
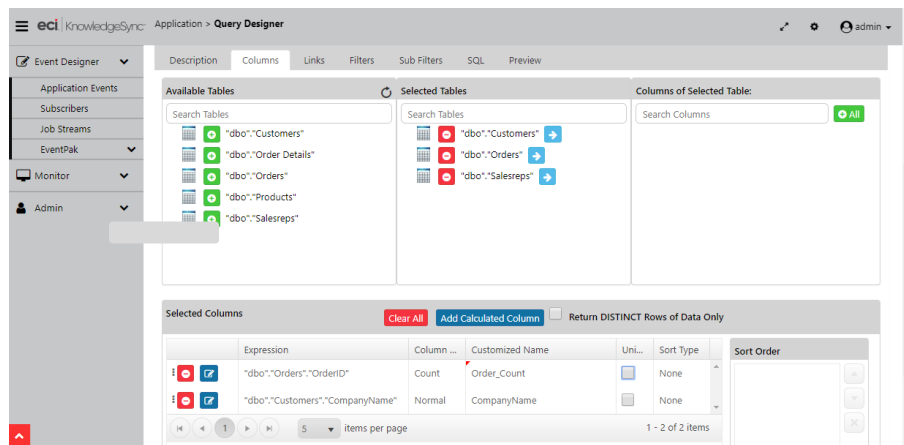
AGGREGATE "COUNT" FUNCTION:
NUMBER OF LATE DELIVERIES PER CUSTOMER

3.11.3.8 Additional Column Selections

Here are a few tips for choosing additional columns in your aggregate query:

Make your "group by" field as informative as possible. For example, to display total sales per salesrep, do not make your "group by" field the salesrep's ID, but rather the salesrep's full name.

To *send* the results of a query to the individuals that are identified *within* the query, include the column that contains the delivery address that the alert will be sent to (e.g., include the salesrep's email address as a query column).

A single query can have multiple aggregate fields. E.g., a query could count the number of jobs last month and show the total cost of those jobs by using both the "count" and "summarize" functions.

3.11.3.9 The "Unique" Field

The "Unique" field does not apply to aggregate queries; you may skip specifying a unique column for an aggregate query.

3.11.3.10 Aggregates and Calculated Fields

Calculated fields may be used with aggregate queries in the same manner that they are used with record-level queries. For example, you could create a column that uses a calculation to identify the weighted value of sales opportunities and then use the "summarize" function to total that value.

3.11.3.11 Sort Type / Sorting Order

(See "Sorting", earlier.)

3.11.4 Filters

Aggregate queries use filters to determine which records are to be included in the aggregate calculation. For example, a query that checks to see if there are more than 6 open, high-priority support calls uses filters to limit the potential records to only those calls that are <u>open</u> and have a <u>high priority</u>.

In this regard, filters for aggregate queries are handled in the same manner as filters for record-level queries.

Aggregate queries <u>cannot</u> use the "If Column Values Have Changed" function.

3.11.5 Sub-Filters

Sub-filters are used <u>only</u> with aggregate queries; it is the sub-filter that <u>tests the result</u> of the aggregate calculation to see if it meets certain criteria.

For example, in an aggregate query that "counts" the number of open, high priority support calls to see if there are <u>more than 6</u>, the sub-filter is what checks to see if the number of calls exceeds 6. (And it's the query's standard filters that tell KnowledgeSync to look only at calls that are "open" and "high priority".)

The following queries have their sub-filters highlighted:

Total sales of *less than $10,000* per region
Average discount percent of *more than 15%*
More than 10 open support calls per agent

Sub-filters configured as follows:

Click on the "Sub-Filters" tab.
Column Name. From the list of "Available Columns to Filter" select the aggregate column you created earlier.
"Is" or "Is Not". Defaults to "is"; click and choose "is not" to change.
Operator. See "Operator Notes" earlier for details.
Compare Value. Specify a constant value, a database field name, a date substitution value, or the question mark. (See earlier in chapter for details.)
Type. The type of data the aggregate column contains. Leave with its default unless comparing against another database field or calculated value (use 'literal').
"And / Or". Since a query can include multiple sub-filters, this gives you the ability to specify "and/or" logic in the sequence of your sub-filters.
Optional Prompt. Used only with a sub-filter that is parameter driven (has a '?' in the Compare Value field). See the section titled "Parameter-Driven Filters" for details.
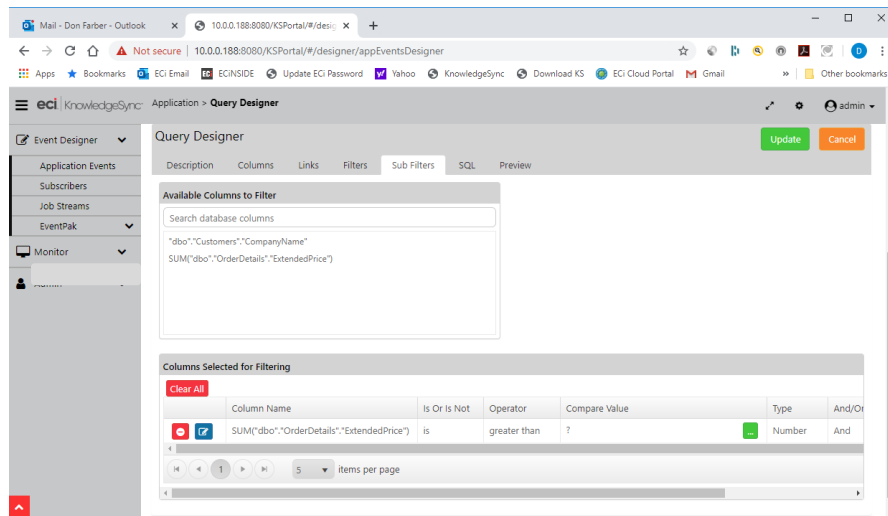
AGGREGATE QUERY SUB-FILTER

3.11.6 SQL Tab (for Aggregate Queries)
See "Query SQL", earlier.

3.11.7 Preview (for Aggregate Queries)
Previewing an aggregate query is the same manner as for a record-level query:
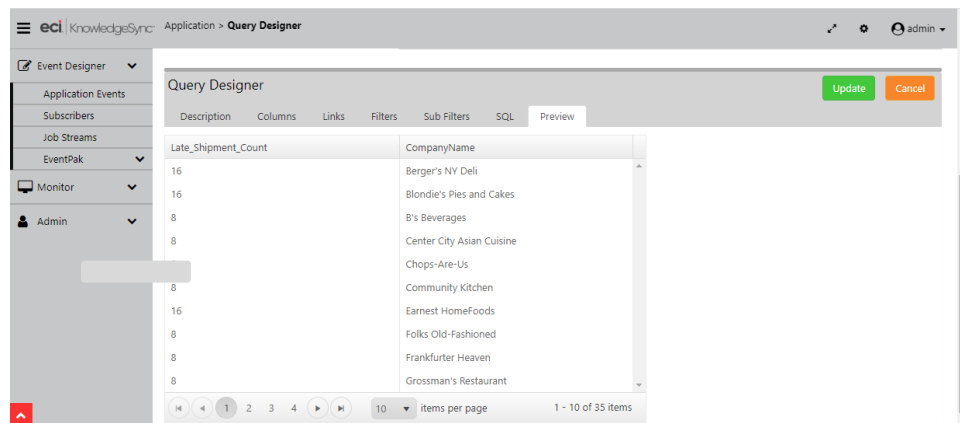
AGGREGATE QUERY PREVIEW MODE

3.12 Linked Queries
A single event can use multiple queries. Consider the following two examples:

Check for more than 20 open service tickets <u>and</u> more than 3 support reps absent from work today.
Check for clients whose orders totaled less than $5,000 over the last month <u>and</u> called into support more than 5 times during that timeframe.
Check for clients who have open sales in a CRM app <u>and</u> are on credit hold in an ERP app.

Each of these events requires two queries each.

The first example requires no "linking" between the queries, because the result sets have no data in common with each other. In such cases, you simply create each query independently and associate both queries with a single event.

(When multiple queries are associated with an event, KnowledgeSync automatically assumes an "and" correlation between them.)

In the second and third examples, however, the queries must be **linked** to each other because the result sets do have data in common – the "client". This requires additional query design considerations.

Data-change queries cannot be linked to any other queries.

3.12.1 How Queries are Linked

If you determine that you need two (or more) queries to identify whether a business condition exists, the first thing to decide is whether the queries will have some **data in common** with each other. Consider the example given previously:

Check to see if a client has purchased less than $5,000 over the last month, and has called into support more than 5 times.

This scenario requires one aggregate query to "sum" a client's sales, and a second aggregate query to "count" the client's number of support calls.

What makes this scenario unique is that both queries need to retrieve their results for **the same clients** – and the "client" is referred to as the queries' "linking field". Here's how to "link" these queries:

Identify the linking field ("client ID").
Decide which query will go first.
Design query #1 (e.g., "Clients Purchasing < $5k")
Go to the "Columns" of query #1 and be sure to include the linking field as one of the columns for this query.
Modify the **Customized Name** of the linking field to something memorable (e.g., "client_link").

3.12.1.1 Design the "Link to" Query

The second query should identify that it is a "link to" query, so give it a description such as "*Clients with > 5 Support Tickets (Link To)*". Design this query as per its requirements and then (to create the link with query #1) follow these steps:

Go to the "Filters" of query #2.
Add a filter on the linking field (e.g., the "Client ID" field).
Select an operator of "is equal to".
In the Compare Value field, key in (within curly braces) the underline customized name of the linking field from query #1 {client_link}.

By designing your "link to" query in this manner, it will automatically retrieve records for the same client as query #1. The following is an example of this:
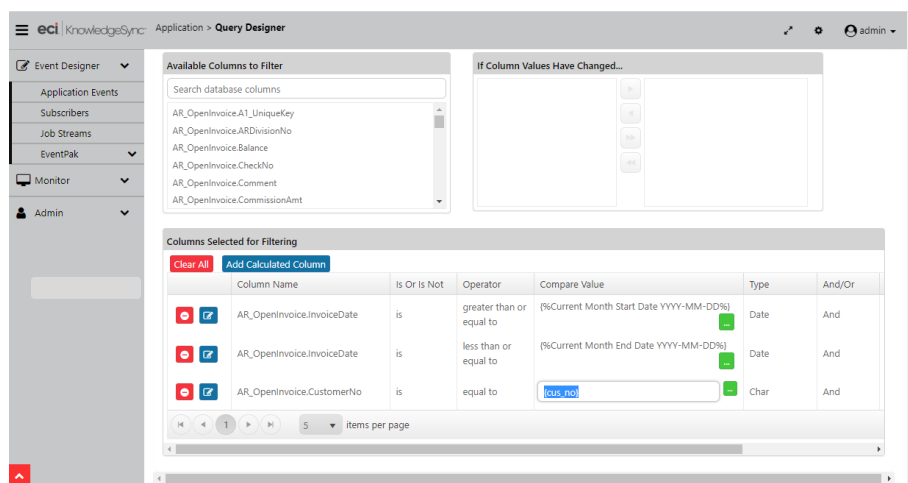
"LINK TO" (COMPARE VALUE) LINKING SYNTAX

Please note of the following:

A "link to" query will execute successfully <u>only</u> if it follows a preceding query containing the appropriate customized name. Do <u>not</u> use a "link to" query stand-alone in an event; it will not execute successfully.

You can have up to 99 linked queries in a single event.

A link-to query should not use a "group by" value. For example, if the link-to query "counts" the number of activities for a specific client, the query should have only a single column (the field to be counted) and the filter will perform the linking function.

The link-to query cannot be used to retrieve "advanced" subscriber information. This information must be retrieved by the initial query.

You can link queries from the same application or from different applications; the technique is the same (identify a "linking field".)

3.12.2 Data Manipulation between Linked Queries

One of the special benefits of linking queries is that it enables you to use data from query #1 in query #2. Consider the following example:

> *When a sales opportunity is placed in a CRM application, you want KnowledgeSync to take the value of that sale, add it to the client's current balance in an ERP application, and then see if that total is greater than the client's credit limit.*

When you link queries together, data from the first query can be used in the second query's "Columns", "Filters", and "SubFilters". Like the linking field, all fields are referred to using their customized name within curly braces, as shown below (note format of column name selected for filtering):
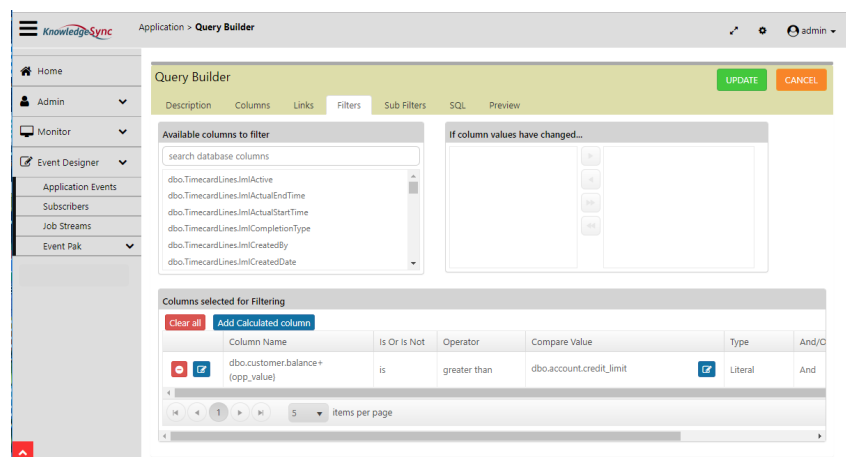
DATA MANIPULATION BETWEEN LINKED QUERIES

> *A field from a previous query needs to be referred to using its <u>customized name</u> enclosed within <u>curly braces</u>.*

> *When an event uses multiple queries, <u>all</u> of the data fields (from all <u>preceding</u> queries) are available to be used in a subsequent query, and all data from all queries is available to be used in the event's alert messages, report selection criteria, and event workflow.*

3.13 Database Triggers

In addition to using queries to identify database conditions, KnowledgeSync supports the use of web service calls (via a REST API) as well as database triggers (which are embedded within external applications) to know when records are added, changed, or deleted.

Please use the KnowledgeSync on-line knowledgebase (or contact KnowledgeSync Support) to get examples of web service calls as well as database triggers and their use in KnowledgeSync events.

4 Subscriber Maintenance

4.1 Event Subscribers

A subscriber is a person (or a computer) that is eligible to receive information (e.g., alerts, reports, charts) from KnowledgeSync. Subscribers may receive notifications via email, text message, web browser, and FTP/SFTP.

Subscribers can be set-up within KnowledgeSync, they can be imported from external applications, or they can be dynamically accessed from any applications external to KnowledgeSync.

4.2 Who Can Be a Subscriber?

Although alerts are often sent to internal employees, KnowledgeSync is also widely used to send alerts to clients, business partners, vendors, suppliers, and prospects.

4.2.1 Using KnowledgeSync for Target Marketing

Since KnowledgeSync can send alerts to customers and prospects just as it can notify employees, consider using KnowledgeSync for direct marketing efforts.

Consider the following scenarios:

Notify a client when they are within 'x' dollars of a bonus program
Notify a prospect about a limited-time offer about to expire
Notify all clients using product 'x' about a new add-on module 'y'

These are all examples of events whose purpose is to keep clients and prospects better informed about marketing-related activities within your organization.

4.2.2 Designating a Computer as a Subscriber

In addition to setting up people as subscribers, KnowledgeSync also allows you to designate computers as recipients of information when an event is triggered.

Using the Copy/FTP/SFTP delivery methods, you can instruct KnowledgeSync to move certain files and/or reports to specific server, disk and directory specifications on one or more computers when an event is triggered.

4.3 How Are Subscribers Categorized?

You may wish to categorize subscribers by department or similar areas of work. This is done through "subscriber groups." For example, you might create such groups as:

Engineering
Customer Service
Executives
Sales
Clients in the State of California
Prospects

You can also create "folders" of subscribers (in which you can create groups), such as:

Engineering Group   (a folder)
Development   (a group)
New Products   (a group)
Maintenance   (a group)
QA   (a group
Doc   (a group)

Using subscriber groups, you can configure an event to alert all the members of one or more groups.

A single event can send alerts to any combination of individual subscribers and groups of subscribers.

4.4   Standard versus Advanced Subscribers

This chapter concerns the configuration of "standard" subscribers – that is the people, computers, or groups of people whom you identify by name when choosing an event's alert recipients.

For example, when a support ticket becomes overdue for resolution, you might wish to notify the Support Manager, Alex Smith. Alex would be considered a standard subscriber because he will always receive an alert when a ticket becomes overdue.

"Advanced" subscribers refer to people who are associated with (or related to) the records that are triggered by a given event.

While an overdue ticket should always notify Alex, you might also wish to alert the customer whose ticket is overdue, as well as the technician to whom the ticket is assigned. These recipients depend entirely on which tickets are overdue and are referred to as an event's "advanced" subscribers.

4.5   Alert Methods & Addresses

Each subscriber can be configured to receive alerts via four methods:

- Email
- SMS (text message)
- Web dashboard (called "webcast")
- Copy/ FTP/SFTP

Alerts can be sent to any devices that support the above communications methods.

Each subscriber may have up to two (2) delivery addresses per delivery method so that they can have different delivery addresses *based on the time of day and day of the week*.

For example, John Smith of "Acme Inc" may wish to receive email alerts at smith@acme.com if an alert happens between 8 AM and 5 PM on Monday through Friday, but he may wish to receive email alerts at jwsmith@gmail.com if an alert happens outside of those days and hours.

> *Note: When an event is triggered and reports run outside of a subscriber's configured time frame, email messages are not sent to the subscriber. For example, if a subscriber has their primary email address configured from 8 am. to 5 p.m., an email message triggered at 7:30 is not sent to the subscriber. When this occurred, the user was not notified of the missed email. Now, a message is added to the log file that tells the user that the email notification was not generated because it was sent outside of the subscriber's allowed schedule.*

4.5.1   Dashboard Alerts (Webcasting)

Dashboard alerts (or "webcasting") is the ability to dynamically generate (and dynamically update) web pages containing alerts, reports, and related information. This component requires its own configuration and is detailed in the "Webcasting" chapter of this guide.

(Also please see the "Dashboard Setup Kit", downloadable from the KnowledgeSync website.)

4.6   Subscriber Entry Methods: Overview

There are three ways that subscribers can be configured:

Manual entry
Subscriber import
Subscriber linking

If you have subscriber data residing in one or more database tables, you can use a KnowledgeSync query to import that data into the subscriber database. Imported data typically includes a subscriber's name and delivery addresses.

V2024-12-16 For Customer Use   KnowledgeSync is an ECI Software Solutions, Inc. technology.   67

Subscriber "Linking" is similar, except that instead of <u>transferring</u> data into KnowledgeSync, "linking" allows KnowledgeSync to read that data in real-time from a table in another database.

4.7   Preparation

If you are manually entering subscriber data, gather the following information for each subscriber:

Subscriber's name, company name, and title
Subscriber's primary and secondary email, text, and FTP addresses
The days of the week (and hours of the day) when the subscriber will receive alerts via their primary and secondary addresses.

If you are configuring subscriber records via Import/Link, to do the following:

Make sure that the database that includes the subscriber data is defined as an "application" within KnowledgeSync.
Identify the fields in the external table containing the subscriber's name, company, title, and primary and secondary email, text and FTP addresses.
Identify the days of the week (and hours of the day) when the subscriber will receive alerts via their primary and secondary addresses.

4.8   Subscriber Maintenance

Log into KnowledgeSync and expand the "Event Designer" branch and select "Subscribers". Expand either the "Computers" or "Users" branches to see your existing subscriber configuration.
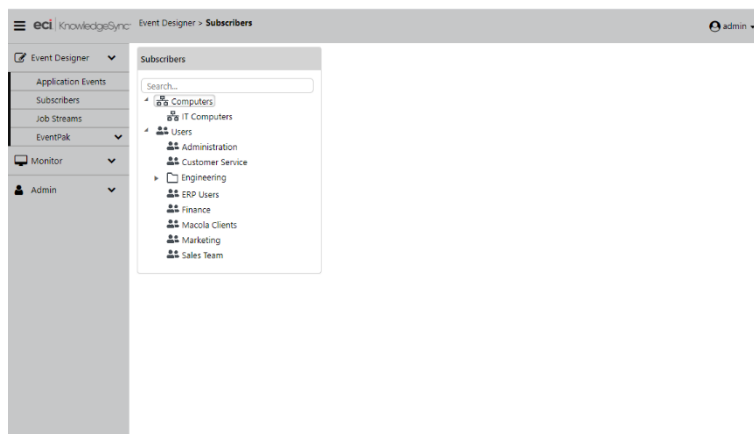
"SUBSCRIBERS" BRANCHES

4.8.1   Group & Folder Maintenance

If you are importing and/or linking subscribers, each "imported" or "linked" collection of subscribers must be associated with a unique subscriber group.

You can create subscriber "groups" (which contain one or more subscribers) as well as subscriber "folders" (which contain one or more subscriber groups or sub-folders).

To add a subscriber group or folder to the top-level "Users" branch, single-click on that branch and select "New Group" or "New Folder". To add a group or folder beneath any other branch, single-click on the branch beneath which you'd like to add a group or folder and click on the corresponding buttons.

Only <u>users</u> may be added to a group and that only <u>groups and folders</u> may be added to a folder.

4.8.1.1   Changing a Group or Folder Name

To change the name of a subscriber group, single-click on the group name and then on the "Properties" button.

To change the name of a subscriber folder, single-click on the folder name and then (on the right-hand side of the window) modify the folder's name.

4.8.1.2   Deleting a Group or Folder

Before deleting a group, make sure that no events are configured to send alerts to anyone within that group or to the entire group.

To delete a subscriber group or folder, single-click on the desired group/folder and click on the red "Minus" button.

If a group is using subscriber linking, you must remove the link before you can delete the corresponding group.

4.8.2   Manually Entering Subscribers

To add a subscriber, follow these steps:

Navigate to the group to which you wish to add the subscriber.
Click on the "New User" button; you'll see the following:
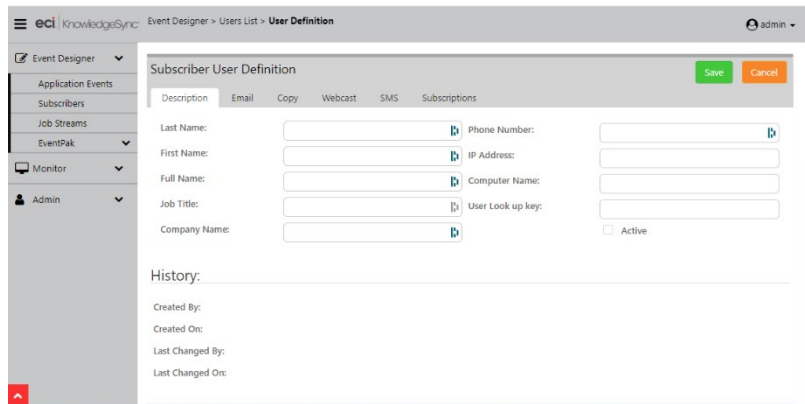
SUBSCRIBER DESCRIPTION

Fill in the user's name, company, title, and related info. If this user is able to be selected as an alert recipient, click on the "Active" checkbox. **Make sure that the "Full Name" has a value, as this is displayed elsewhere in the application.**

If the subscriber is a computer, fill in the name, description, and IP address.

See the following section for details on the use of the "User Lookup Key" field.

Use the Email, Copy, Webcast and SMS tabs to specify the alert delivery addresses for the subscriber.
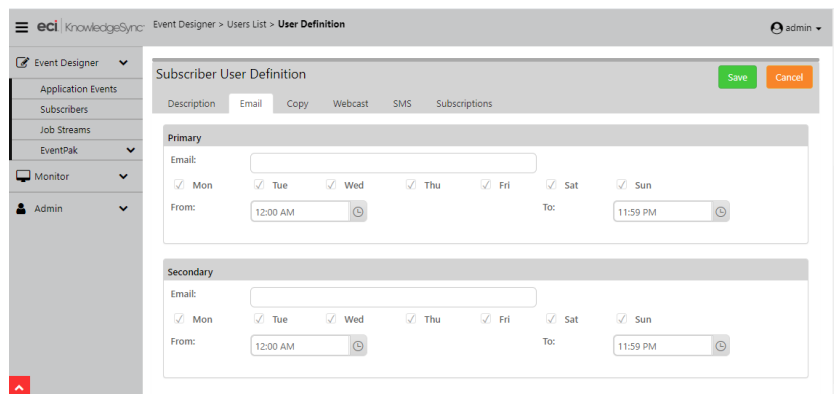
SUBSCRIBER EMAIL ADDRESSES

You can specify which address (primary or secondary) will be used based upon the time of day and day of the week when an alert occurs. Click on the "Save" when done entering the subscriber record.

4.8.3   The "User Lookup Key"
Consider the following scenario:

You have an application that tracks calls assigned to reps. Each rep is identified by a four-digit code such as '1234'; however, the application itself does <u>not</u> store reps' email addresses.

When an alert is triggered, KnowledgeSync needs to look at the triggered record, use the rep's code to <u>look up</u> the corresponding subscriber record in KnowledgeSync, and then send the alert to the address associated with that subscriber record. That's what the "lookup key" is for.

In the above example, the "User Lookup Key" would contain a rep's four-digit code. It would then be used to link a user's "code" in an application with a specific subscriber record in KnowledgeSync.

The "user lookup" functionality works with user groups as well; instead of looking for a match between a database field's content and a user's "lookup key", KnowledgeSync looks for a match between a database field's content and the <u>name</u> of a user group. If such a name is located, KnowledgeSync sends alerts to all members of that user group.

4.8.4   Copy/FTP Notes
The "copy" method (Copy, FTP & SFTP) method lets KnowledgeSync move files and reports to specific locations (server, disk & directory).

SUBSCRIBER "COPY" TAB

The copy location should be specified using standard UNC format (as in "//computername/sharename"). You may also specify an FTP/SFTP address; do not reference any mapped drives in your location specifications and note the fields for username and password credentials, if required.

4.8.5   Webcasting Notes
Please refer to the chapter titled "Dashboard Alerts (Webcasting)" for details.

4.8.6   SMS Notes
The "SMS" method lets KnowledgeSync send alerts via text message; KnowledgeSync uses Twilio as its method for sending text messages, and you need to sign-up for the Twilio text messaging service in order to receive alerts via text message. (See https://www.twilio.com/ for details.)
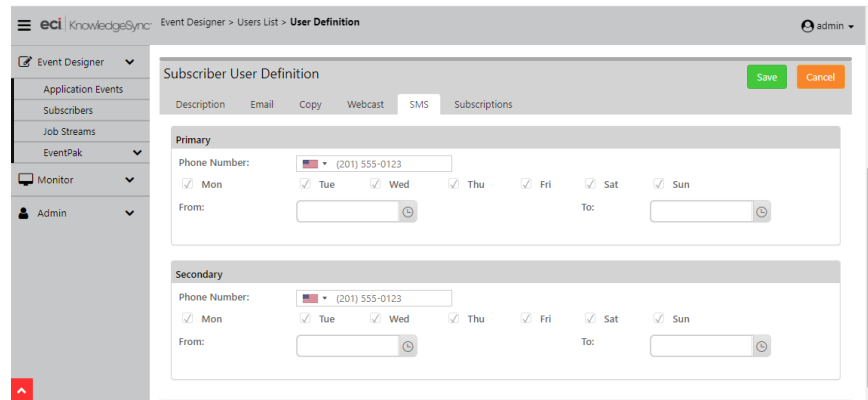
SUBSCRIBER "SMS" TAB

You will need to purchase a Twilio account in order to send text message alerts via this method; see the "Administration" chapter of this guide for details on configuring your text messaging sending account.

Depending on the texting provider, you may also be able to deliver text messages by sending an email message to a specific provider's email address along with the recipient's cell phone number.

4.8.7 The "Subscriptions" Tab
The "Subscriptions" tab is an read-only listing of all the events that a subscriber is currently enrolled to receive alerts about.

4.8.8 Creating a Template Subscriber Record
When manually entering subscriber records, it is suggested that you create a subscriber template and then copy that template for each user you wish to add.

To create a template, simply add a new subscriber and give it a name such as "User Template." Modify the fields to reflect the common values that should be used in all new users, and then save that record.

The next time you need to create a new subscriber, simply copy that template instead of clicking on the "New User" button. (See "Copying Subscribers".)

4.8.9 Copying Subscribers
To copy a subscriber record, highlight the subscriber you wish to copy, click on the "Copy User" button followed by the "Paste User" button.

A duplicate of the subscriber will now appear. Go into either of the two subscribers and modify it as needed.

4.8.10 Deleting Subscribers
Before deleting a subscriber, make sure there are no events that send alerts to them. (Open the subscriber record of the subscriber to be deleted, and review their "Subscriptions" tab.)

To delete a subscriber, single-click on the subscriber you wish to delete and click on the red "Minus" button.

Note that there is no "Undo" option once you have deleted a subscriber.

4.9 Importing / Linking Subscribers
The configurations for importing and linking subscribers in KnowledgeSync are very similar; the only difference is that the import option transfers data from an external application into KnowledgeSync, and the link option creates a "live link" between subscriber data in an external application and KnowledgeSync.

If you have the option to choose one method versus the other, choose the "linking" option, as it is dynamic, and does not cause the same data to be stored in multiple locations.

4.9.1   Create the Import/Link Query

In order to import/link subscriber data, you need to first create a KnowledgeSync query to retrieve that data. The data that the query should retrieve (if available) includes:

Subscriber full name, last name, and first name
Subscriber title, company name, and phone number
Subscriber's primary and secondary email addresses, preferred copy locations, and SMS addresses.

Your query should be designed just like any normal, record-level query; typically it retrieves data from just one table that contains subscriber data. It may (or may not) include filters, and it must have a "unique" column identified.

4.9.2   Create the Import/Link Group

For each group of subscribers that you will be importing or linking to, you should create a subscriber group (e.g., "ERP Users").

An import/link subscriber group may also contain manually added subscribers.

4.9.3   Access the Import/Link Function

Follow these steps:

Single-click on the group that will contain the imported/linked subscribers.
Click on the "Properties" button.
Click on the "Subscriber Linking" button.
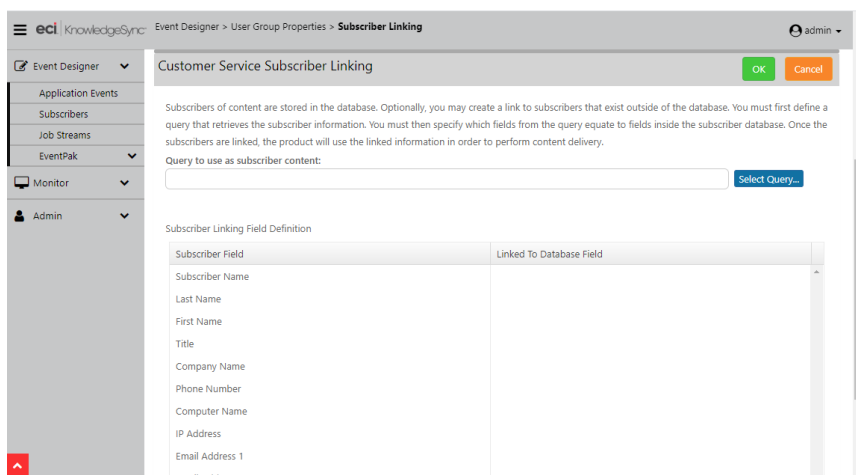
SUBSCRIBER LINKING OPTION

4.9.4   Choose the Import/Link Query

Click on the "Select Query" button. Select the KnowledgeSync application your subscriber linking query is in. Once you select the appropriate query, you can proceed with selecting the "linked" subscriber data.

The left side of the window displays KnowledgeSync's subscriber data fields. The right side lets you select which fields from the selected query will be mapped into the corresponding subscriber data fields.

For each subscriber field you wish to be populated with data from your query, click in the corresponding "Linked to Database Field", and choose the query field whose value you wish to map. Click on OK when done.

The same query field may be mapped to multiple fields within the subscriber database (such as mapping a database's single field for "email address" to both "email address 1" and "email address 2" in KnowledgeSync.

Your completed mapping will look like the following:
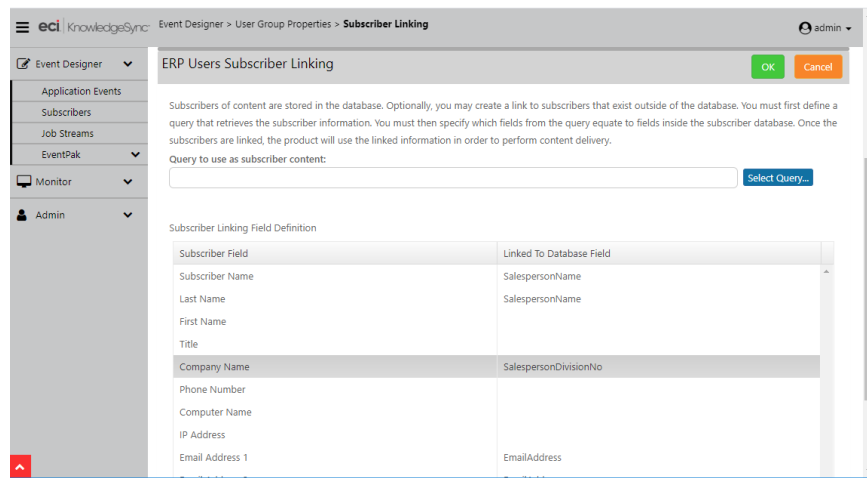
SUBSCRIBER LINKING FIELD MAPPING

The "<u>Full Name</u>" field <u>must</u> have a value mapped to it.

When you save this information, KnowledgeSync executes the import/link and retrieves the corresponding subscriber records from your external database. These records are displayed as subscribers for the corresponding group.

4.9.5   Choose "Link" or "Import"

Once you finish mapping your fields and save your work, KnowledgeSync assumes that these subscribers will continue to be retrieved via a dynamic link.

This means that KnowledgeSync is <u>not</u> importing this data, but rather is reading it dynamically out of an external database. You may leave this link intact or you may change it to an "import" where this subscriber data is physically stored in the KnowledgeSync database.

Imported subscriber data <u>may</u> be maintained & modified within KnowledgeSync; linked subscriber data may <u>not</u> be modified within KnowledgeSync.

To change a linked subscriber group into an import of subscriber records, select the subscriber group for which you wish to disable linking, select "Properties" and then click on the "Remove Subscriber Link" button.
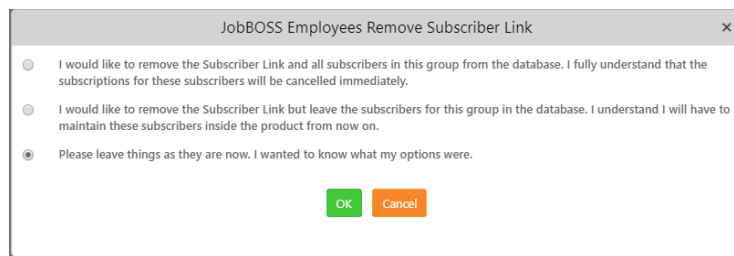
SUBSCRIBER LINKING OPTIONS

Click on the <u>second</u> option ("I would like to remove the subscriber link but leave the subscribers for this group in the KnowledgeSync database.")

Click on "OK". The "link" will be removed and the subscriber data retained.

You must **save** your subscriber link (and thus import the subscriber records) before removing the link itself.

4.9.6   Removing a Subscriber Link

When you remove a subscriber link, you may retain the details of individual subscribers, or you may remove both the link and subscriber details. To remove both the link and the related subscriber data, select the subscriber group for which you wish to remove the link and its subscribers, select "Properties" and then click on the "Remove Subscriber Link" button

Click on the **first** option ("*I would like to remove the subscriber link and all subscribers in this group from the KnowledgeSync database.*")

4.9.7   Editing Imported/Linked Subscribers

If you import subscribers (i.e., remove the link but retain the subscribers), you will be able to edit and modify those subscriber records. If you create a group of linked subscribers, however, only the alert hours and days will be available for editing.

4.10 Intelligent Subscriber Groups

One of the benefits of creating subscriber groups using queries is that you can create "intelligent" groups of subscribers. For example:

Clients in the state of California
Prospects in the banking industry
Suppliers who have pending backorders

All these groups utilize queries that have filters within them so as to retrieve a specific sub-set of records.

5   Event Designing

An event checks to see if certain conditions exist and (if so), the event executes one or more "deliverables", such as sending alerts, generating and delivering forms, documents, and reports, and executing workflow actions, such as updating applications, calling web services, and so on.

5.1   How an Event Works

Most KnowledgeSync events run on a schedule, such as hourly, daily, weekly, or every 'x' minutes. An event may be configured without any "triggers" so that it always runs on its schedule (such as a weekly newsletter), or an event may use one or more triggers to determine if certain conditions exist, and (if so) execute the corresponding deliverables. (An event may also be triggered by the receipt of incoming email.)

The deliverables that an event can execute may include the sending of alerts, the generation and delivery of reports, and the execution of workflow, which can include application updates and web service calls.

5.2   Event Triggers

Triggers enable an event to determine whether specific conditions have been met. Triggers fall into one of three categories:

Database queries (.NET data provider, ODBC & OLE/DB)
Web API (via a REST API web service)
Visual Basic script

Triggers are used to identify conditions within application databases, incoming e-mail messages, websites, and the operating system.

A single query may be used in one or more events. For example, a trigger designed to retrieve orders of more than 'x' dollars could include be used in one event looking at orders over $100 and a second event looking at orders over $500.

5.3   Event Alerts

Each event may have its own alert message content. An event's message content may include data from the corresponding conditions that have caused the event to trigger.

KnowledgeSync's alert delivery methods include:

Email
SMS (text message)
Web browser (aka "dashboard alerts")
Copy / FTP / SFTP

5.4   Event Files, Forms, Reports, & Charts

As part of the message content, an event may generate and deliver forms and documents (such as invoices or statements), the contents of files (such as Excel and PDF files) the contents of one or more reports (Crystal & SSRS), and graphic displays (pie charts & bar charts).

5.5   Event Actions (Workflow)

An event can execute workflow actions to:

Add or update application database records
Move data from one application to another
Push or retrieve information from websites
Trigger site-specific programs or procedures

5.6   Event Subscribers

The recipients of an event's alerts are referred to as subscribers. Each event may send alerts to individual subscribers, to groups of subscribers, and/or to one or more computers designated for delivery.

Additionally, events may be configured to notify one or more people who are "related" to the triggering condition (e.g., notify the technician to whom a service call is assigned).

5.7   Event Schedule

KnowledgeSync events are "scheduled" to periodically (and automatically) check to see if the corresponding conditions have been met.

Each event has its own schedule; very time-sensitive events may have a very frequent schedule (e.g., every five minutes), whereas less time-sensitive events may be checked on a less frequent basis (e.g., daily, weekly, monthly).

5.8   Checked vs. Triggered

A "checked" event indicates that KnowledgeSync has successfully <u>run</u> the event. The conditions may have proven true, or they may not. An event that is "triggered" is one that has been checked and whose <u>conditions have proven true</u>.

5.9   Monitoring the Status of Events

The KnowledgeSync Monitor tracks the status of the event's execution:

Events that are pending submission
Events that have been checked but not triggered
Events that have been triggered
Events with errors and have failed to run successfully

If an event errors because of an error in its configuration, the event goes into a "pending" state and will be automatically re-queued for submission. This means that you can modify the event (or its query) and KnowledgeSync will automatically re-run the event and use the modified version of it.

> *The Monitor includes details and audit tracking of all an event's deliverables – its alerts, generated forms, documents, and reports, and executed event workflow (e.g., adding or updating records in application databases).*

5.10 An Event without a Query

Consider the following scenarios:

Every Monday at 8:45 AM you want to remind managers to attend the 9:00 AM staff meeting
Daily at 5:00 PM you want to transfer an "orders" file to a database server

These scenarios are examples of events <u>without a query</u>; they are scheduled to run on a re-occurring basis.

5.11 Before You Design Events

Make sure you have done the following:

Reviewed query design
Added yourself as a subscriber
Defined your email sending account (in the Admin component of KS)

5.12 Accessing Event Design

Log into KnowledgeSync, click on the **Application Events** branch, expand the application you wish to work with, and then click on the "Events" branch.

The following window opens:
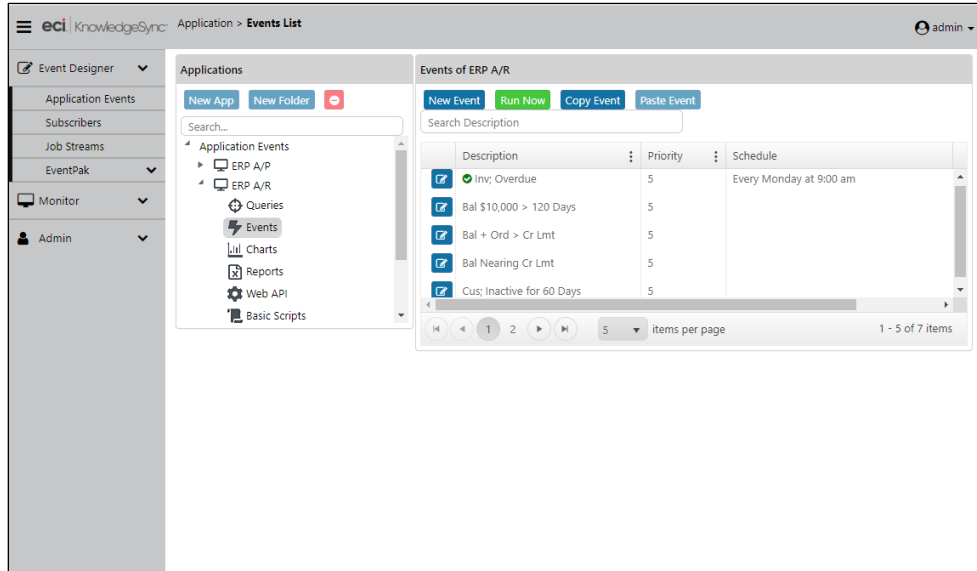
EVENT LISTING FOR AN APPLICATION

5.12.1 Create, View, Edit, Delete, Copy, or Run an Event

To create an event, click on the **New Event** button.

To view or edit an event or its components click on the blue edit button to the left of the event name.

Use the red minus sign at the far-right of an event to delete it.

Use the **Copy Event** and **Paste Event** buttons to make a copy of an event. (And don't forget to change the name/description of the copied event!)

If you wish to run an event once, <u>right now</u>, highlight the desired event and then click on the green **Run Now** button.

5.12.2 An Event's Components

When you create a new event (or open an existing one), you'll see five sections on the left side of the event configuration window:

- **Event**: The name and general settings of the event.
- **Schedule**: How often the event is to run (check for its conditions and trigger its responses).
- **Triggers**: The conditions that cause an event to execute its alerts and other response actions.
- **Deliverables**: The alerts, reports, charts, and files associated with the event, the manner in which that content will be delivered, and the workflow actions that will be executed when the event's conditions are met.
- **Subscribers**: The users, groups, and/or computers that will receive alert content.

These components are identified by the shaded headings on the left-side of the following window:
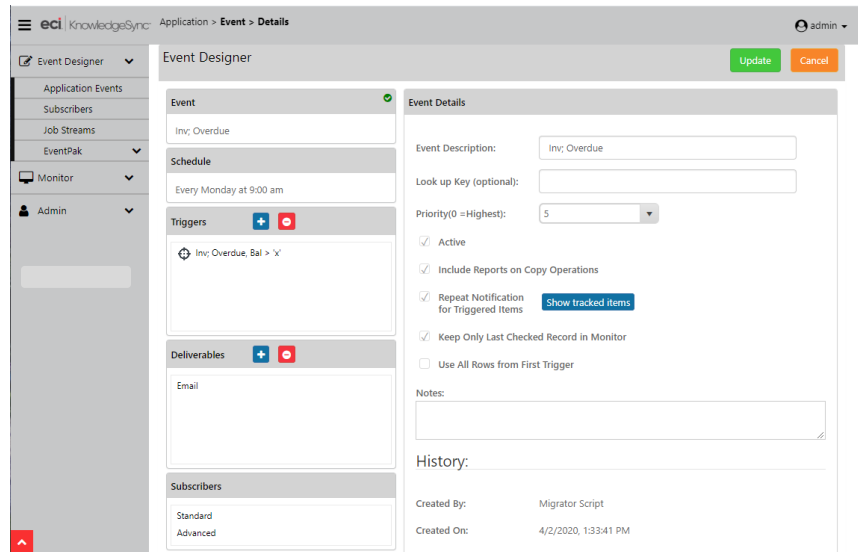
THE 5 COMPONENTS OF AN EVENT (ON THE LEFT)

5.12.2.1 Event Summary

When you create a new event (or open an existing event), you'll see the "Event Details" displayed on the right-side of your window as shown in the previous illustration. The event details window contains the following information:
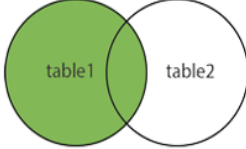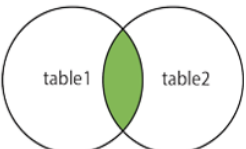
- **Event Description**: Use a standard naming convention, such as "Subject; Condition", as in "Accounts; Overdue".
- **Lookup Key**: Leave blank, unless an event's execution is to be "called" from an external application. (See "Utilities" chapter for details.)
- **Priority**: The order in which events <u>with identical schedules</u> (e.g., daily at 9 AM) will be submitted. Higher priorities ('0' being the highest) will be submitted first.
- **Active**: Whether an event is eligible to be scheduled (run) by KnowledgeSync.
- **Include Reports on Copy Operations**: If the event includes any reports (Crystal or SSRS) or charts, this option lets you control whether those reports and charts will be eligible to be "copied" (or ftp'd) to the recipients' preferred location.
- **Repeat notification for triggered items**: Whether this event will <u>repeat</u> alerts for a record that meets this event's criteria. See following section for details.
- **Keep only last checked record in Monitor**: Whether to retain the history of every time the event has run, or retain the history of only the <u>last</u> time the event was run.

  KnowledgeSync keeps the history of every time the event has been <u>triggered</u> – that is, whenever the event's conditions were met. The suggested setting for this field is <u>checked on</u>.

- **Use All Rows from First Trigger**: This pertains only to events with multiple triggers and allows you to specify that only the first trigger returns all records that meet the event's criteria, as opposed to all triggers cumulatively retrieving all matching records. The default setting for this field is checked ON. See the next section for further details.
- **Get All Results**: All data from multiple triggers are retrieved. KS returns all results from all rows from all triggers.

5.12.2.2 Multiple Event Triggers

If you have multiple triggers, KnowledgeSync uses the results from the first trigger as the master table for the event. Results from the subsequent triggers are treated like an SQL "inner join" or "left join" based on your "Use All Rows from First Trigger" selection. KnowledgeSync will not add rows to the table in subsequent triggers, only modify and/or remove them.

Use All Rows from First Trigger	Example: Two triggers are required to get customer data.
Checked Behaves like a "left join": 	First trigger gets the customer list: 	Customer ID	Customer Name
1001	Built Rite
1002	Process Perfect
1003	Create LLC | Second trigger gets the "Last Order Shipment Status" for each customer: | Customer ID	Customer Name	Last Order Shipment Status
---	---	---
1001	Built Rite	\<none>
1002	Process Perfect	\<none>
1003	Create LLC	Shipped	 *Three rows are processed even though only one "Last Order Shipment Status" was returned.*
Unchecked Behaves like an "inner join": 	First trigger gets the customer list: 	Customer ID	Customer Name
1001	Built Rite
1002	Process Perfect
1003	Create LLC | Second trigger gets the "Last Order Shipment Status" for each customer: | Customer ID	Customer Name	Last Order Shipment Status
---	---	---
1003	Create LLC	Shipped	 *Only one row is present since only one "Last Order Shipment Status" was returned.*

5.12.2.3 Repeat Notification for Triggered Items

Consider an event that runs every hour, such as a "low stock alert". Let's say that an item in inventory causes this event to trigger at 9 AM. This item may still meet the event's conditions when the event runs again at 10 AM. You need to decide if a record that initially meets an event's criteria (and triggers) should (if the record continues to meet the event's criteria) "repeat" its triggering each ensuing time the event runs. This is what "repeat notification for triggered items" is asking you.

For some conditions, you might want an event to trigger underline repeatedly, as long as a record continues to meet an event's criteria. For other conditions, you might want an event to trigger just once for each record that meets its criteria. Consider the following two examples:

A salesrep is notified every day about overdue activities. If an overdue activity remains open for multiple days, you want to alert the salesrep each day. (**repeat notifications**)
Every hour, you check for new orders and send customers an order confirmation. Once a confirmation has been sent about a particular order, you don't want to send another confirmation about that same order. (**do not repeat notifications**)

If you want an event to trigger once for a record meeting an event's criteria, do not place a checkmark in "Repeat notification for triggered items."

If you want an event to trigger repeatedly about a record until that record no longer meets the event's criteria, do place a checkmark in this field.

> *To configure an event to not repeat notifications, the event's trigger must have a "unique" value identified. If the trigger has no "unique" column, the event will automatically operate in "repeat" mode.*

5.12.2.4 How "Triggered" Records Work

When a record is "triggered" by an event, the triggered record will appear under the "**Show Tracked Items**" option for an event. If, the next time the event runs, the same record <u>no longer meets the event's criteria</u>, the triggered record will be automatically removed from the event.

You can manually clear triggered records for an event using the "Delete Selected Items" and the "Delete All Items" options from the "Show Tracked Items" window. (Detailed at the end of this chapter.)

5.12.2.5 Using "Repeat" in a Field-Level Change Event

If an event is checking for a "change" to the value of one or more database fields, the "Repeat notifications" option must be checked <u>on</u>.

5.12.3 Schedule

An event's schedule controls how often the event runs; if an event has no schedule, it will read "add a schedule". Click on an event's schedule to see the following window:
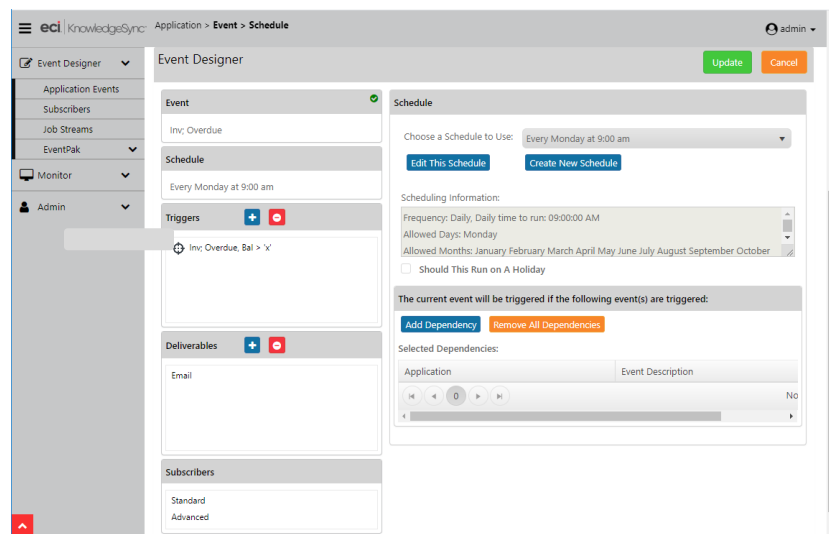
EVENT "SCHEDULE"

From this window you may select a schedule for an event, create a new schedule, or edit the configuration of an existing schedule. KnowledgeSync comes pre-configured with frequently-used schedules; you may add to or modify these. Note the following:

If you wish to <u>test</u> an event, give it a schedule of "none" and then use the option to "Run Now" to submit the event immediately.
The schedule called "**When email arrives**" is for use only with events that are monitoring the content of <u>incoming</u> email messages.
If an event needs to execute in real-time, use a database trigger instead of scheduling the event for periodic submission.
For scheduling frequencies that are <u>not</u> available via the scheduling options, use the event's query to restrict when the event can run. For example, to run an event on "the first Monday of a month", use a schedule of "every Monday" and then use your query to check for a "day number" that falls between one and seven.

5.12.3.1 Creating/Editing a Schedule

When you create or edit a schedule, you'll see a window like the following:

SCHEDULE DEFINITION WINDOW

The definitions of schedules vary based on their frequency, but in general you'll have the following fields to fill out:

Schedule. Type in a description of the schedule (e.g., "Everyday at 3 PM")
Should not run until. Enter a date in the future when events using this schedule will begin to be submitted. Leave blank if this schedule is to be activated immediately.
Frequency. Choose from:
- o   Every 'n' minutes
- o   Hourly
- o   Daily
- o   Weekly
- o   Monthly
- o   Annually
- o   When email arrives
- o   None (used only for events that have no schedule)

Based on the chosen frequency enter the additional scheduling details, such as the time of day to run, the day of the week, et cetera.

When creating or editing a schedule, you may click on the "**Allowed Range**" tab. Specify the months of the year, the days of the week, and the hours of the day when this schedule is eligible to run events. Times are inclusive.

SCHEDULE "ALLOWED RANGE"

Should run on holiday. Check if you <u>do</u> want an event to run on a holiday (holidays are defined in the Administrator module).

5.12.3.2 Event Dependencies

Dependencies let you define scenarios where event 'x' is automatically – and immediately – submitted by KnowledgeSync if one or more <u>other</u> events have been triggered. For example:

If event 'a' is triggered, automatically submit event 'd'.
If events 'a' and 'b' are triggered, automatically submit event 'e'.

You specify a dependency from the "dependent" (or child) event and not from the "parent" event. Thus, if you want "event 'b' to run if event 'a' is triggered", you would go into event 'b', go to the Schedule tab, and make event 'b' dependent on event 'a' being triggered (having its conditions met).

> *With event dependencies, a dependent event ('b') will be submitted as soon as it's determined if event 'a' <u>has been</u> <u>triggered</u>, and <u>not</u> when event 'a' has finished all its processing. Thus event 'a' may still be running some alerts, reports, and actions at the time event 'b' is called.*

If you need to make sure that an event has completed ALL of its processing before submitting a dependent event, please use "Job Streams" instead of event dependencies.

5.12.4 Triggers

An event may use 3 different kinds of triggers:

Queries (.NET database provider, ODBC or OLE/DB). An event may use one or more queries. (See "Query Designing" chapter.) An event may also use a combination of queries and web API calls.
Visual Basic (VB) Script. An event may use a <u>single</u> VB script. (See "Visual Basic Scripting" chapter.)
Web API calls. (See "Web API" chapter.) An event may use one or more web API calls, and an event may use a combination of web API calls and one or more queries.

To add a trigger, click on the blue "Plus" sign in the "Triggers" section of an event. To view or modify a selected trigger, single-click on that trigger. (Once you select a trigger for editing, a blue "Edit" button on the right side of your window will take you into a trigger's configuration.)

To delete a trigger, single-click on the trigger and click on the red "Minus" sign.

When you display a trigger associated with an event, a window like the following will appear:
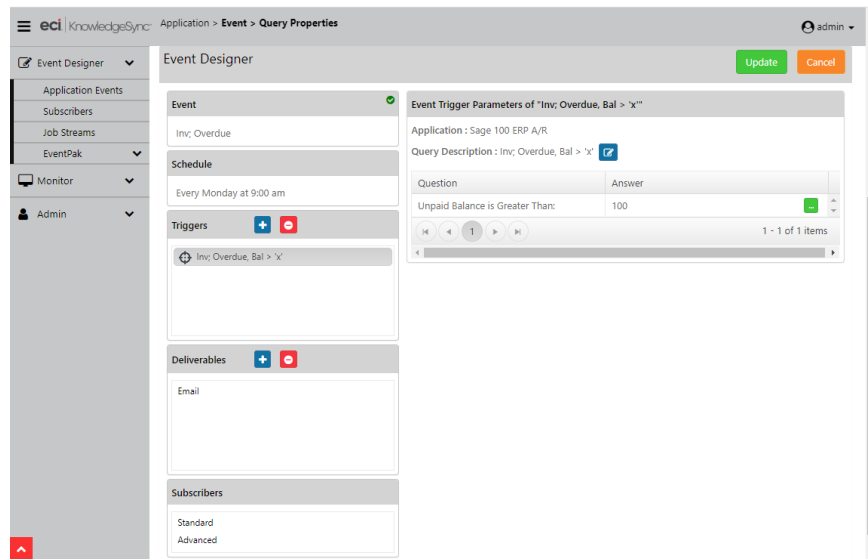
<div align="center">EVENT TRIGGERS (A QUERY)</div>

5.12.4.1 Adding/Removing a Trigger

When you click on the blue "Plus" sign to add a trigger to an event, you'll have the choice to add a query, a VB script, or a web API to the event.

An event that uses a VB script as a trigger cannot use any other triggers along with that script.

If you choose to add a query, you'll be presented with a list of queries from the current application; you can choose to select a query from a different application, if you wish. If the query you select has any parameters, they will appear on the right side of your window, under "Event Trigger Parameters".

(If you add or modify any answers to an event's trigger parameter, be sure to click on either "Update" or "Save" to retain your work.)

If you select multiple queries for an event (or a combination of queries and web API calls), an "and" condition is automatically assumed between them. Note that you can use a combination of queries across multiple applications (e.g., one query based on data in a CRM application, and another query based on data from an ERP application).

An event that uses a query which looks for <u>a change to a database value</u> cannot use multiple queries.

5.12.4.2 Event Trigger Parameters

A trigger may have conditions that are "parameter-driven"; those whose 'compare value' is specified on the event level.

When creating or editing an event, if the event uses a query (with "parameters"), a script (with "global variables"), or web API calls (with "request parameters"), those parameters will be displayed on the right-side of the event design window (under "Event Trigger Parameters"). You can type in (or select from a list, if appropriate) the "answers" to each of the parameters.

Parameters cannot be left blank (or else the event will fail to run successfully).

If an Event Trigger Parameter appears, the parameter will typically begin with a "Question" that instructs you on the type of answer required (e.g., "Please enter a beginning date range").

5.12.4.3 Parameter Answer

Most parameters can be answered by keying in (or selecting) a specific value. You can manually type in the value, or (if available) you can choose a value from a list. KnowledgeSync is case-sensitive, so a compare value of "high" will not retrieve those records that have a value of "High".

To select a value from a list, follow these steps:

- Click in the "Answer" field and type your answer, or;
- Click on the list button and choose from the list (which shows a list of values that have appeared in this field), or;
- Click and select any of KnowledgeSync's date substitution variables.

5.12.4.4 Date Substitution Variables

Consider the following event:

Send an alert if there are any sales of greater than $20,000 due to close next week.

This event would use trigger parameters such as:

Sales Due to Close Starting on?
Sales Due to Close Ending on?

Using "date substitution variables," KnowledgeSync can substitute in "next week's starting date" and "next week's ending date" whenever the event is run. To use a date substitution variable in answer to a trigger parameter:

Click on the list button next to the parameter's "Answer" and review the list of "Available Values for Use" in the ensuing window. The variables are available in multiple formats; variables without any format use your system's regional settings date format.
Choose the value that you wish to use.

Once you select a variable as an answer, you can edit it to append specific values. For example, {%Current Date%} 23:59 means today at 11:59 PM.

5.12.4.5 Using a VB Script Instead of a Query

You may use a visual basic (VB) script to monitor for conditions in databases, or even conditions within your operating system, such as:

Low disk space, full mailboxes, or application errors
Files that have arrived or grown too large

You first must create the VB script that will be used as a trigger; please see the chapter titled "Visual Basic Scripting" for details on the configuration of these scripts.

Once you have created a script for an event to use in this manner, select that script as a trigger for an event.

> *An event can use only one script, and it cannot be combined with any other triggers.*

5.12.4.6 Using Database Triggers

For information on how to use database triggers (such as SQL triggers) instead of a KnowledgeSync trigger, please visit the KnowledgeSync knowledgebase (at the KnowledgeSync website's "Support" page) to download samples and instructions.

5.12.5 Attaching Files to an Alert

There are two ways that files can be attached to an alert:

Via the Deliverable called "File". This allows you to attach one or more specified files to an alert message.
Via the variable "attachment_files" which is referenced in an event's trigger. This variable allows a trigger to dynamically retrieve one or more files based on the results of the trigger. (See the "Query Designing" chapter for more details.)

The variable "attachment_files" may be used in all triggers – queries, VB scripts, and web API calls – to refer to one or more files that are associated with the records being retrieved. If using a query, your query must include a column with a customized name of "attachment_files"; if using a VB script or web API call, your script/call must include the response parameter or variable named "attachment_files".

5.12.6 Deliverables
An event's deliverables are divided into three groups:

- Alert Content (reports, charts, and files)
- Alert Delivery Method & Message (email, SMS text and webcast)
- Actions (create file, run VB script, run Program, submit SQL, and make a REST API call)

5.12.6.1 Monitoring the Status of Deliverables
The KnowledgeSync Monitor tracks the status of all of an event's deliverables (alerts, reports, & workflow actions), including those actions that are pending, have been executed, and those that are in an error state.

If a deliverable fails to complete successfully because of errors, you may (depending on the deliverable) be able to: 1) correct the error within KnowledgeSync, 2) delete the erroring deliverable, or, 3) set the erroring condition to "manually completed". KnowledgeSync will attempt to re-execute an erroring deliverable every minute.

In regards to option #1 above, the KnowledgeSync Monitor module includes an option called "Corrective Actions". This option is available only if the logged-in user has the permission to "take corrective action on monitor screens", in their user profile. If users have this permissions, there will be an "Edit" button on the corresponding "Errors" tab within the Monitor. The function of this button will vary, based on the specific error being encountered.

(E.g., for email errors, you have the ability to update the email address; for file delivery errors, you have the ability to update the file delivery location.)

5.12.7 Alert Message Design
The method by which you design alert messages is identical regardless of the type of alert that you are sending (email, text, webcast). Thus all design options are detailed in the "Email" section of this chapter; the other delivery methods will address only the unique aspects of each.

If an event has one or more deliverables already configured, you will see them listed under "Deliverables" and you may click on any of these deliverables to view or modify them. To add a new deliverable, click on the blue "Plus" sign next to "Deliverables" and then (on the right-side of your window) select the desired deliverable.

5.12.8 Email Deliverable
When you create an email deliverable, you may choose whether the alert message will be in plain text (check the "Plain Text" option box) or in HTML (do not check "Plain Text").

Note: For HTML messages, users need to change browser settings: Browser > Settings > Language > English (US) or English (Australian) (whichever is applicable)

The plain text editor:
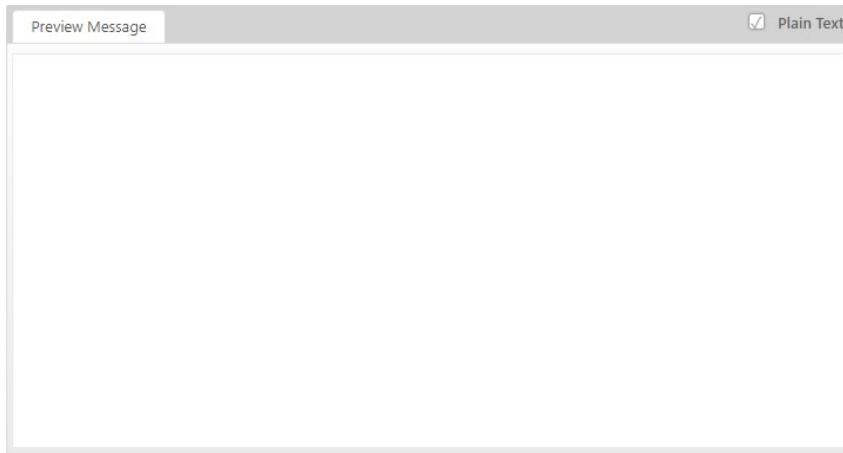
The HTML editor:
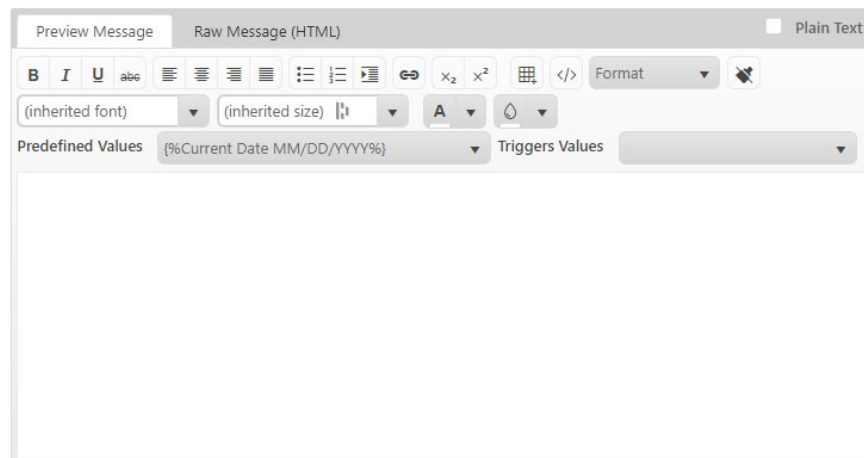
Once in the Email deliverable, the following is available:

Values Available for Message Content. Two scrolling windows here:
- the one on the left is a list of date variables (such as today & tomorrow, American date variables and European date variables), HTML commands, and repeat text commands
- the list on the right displays all the data fields from the event's triggers

These fields may be used anywhere in your alert message. Note that these variables are displayed in the HTML editor and (when working with plain text alerts) those variables are displayed at the top of the Email Deliverable window.

The two HTML commands should be used ONLY if you are creating a plain text alert message AND wish to have part of that message displayed in HTML format. When you use the HTML message editing window, the two HTML commands are automatically inserted around your content, so you don't need to add them manually.

The two repeat commands tell KnowledgeSync to repeat the details of all (or multiple) records in a single alert message. (More on these functions in the following sections.)

Email Account to Send from. This refers to the email account (configured in the Admin option) that KnowledgeSync will send the email alerts from. Click on the list button and choose a corresponding account. An event will not be able to send an email alert if this field is left blank.

Email Reply Address. (Usable only if sending email via an Internet/SMTP method.) You have three choices for this field:

- o **Leave blank**. The default "from" and "reply" name configured in the setup of the email sending account will be used.
- o **Hard-code**, e. g., *Tech Support <support@yourcompany.com>*
- o **Use a database field's value** (e.g., the email address of a sales rep associated with an order). To use such a value, the database field must be one of the columns selected in the event's trigger. (Click on the list button in this field to select from the list of values from this event's triggers.)

The address used **must** be a valid email address. If not, the email alert will fail to be sent.

Message Subject. Enter the subject of the outgoing alert message. You can use any "available values" in the subject field (position your cursor in the subject and double-click on the selected value you wish to appear there). This includes date variables.

Message Text. Enter the content of the alert message; typically, a combination of hard-coded text and field values. The message text may be formatted in either plain text or HTML. If using plain text, be sure to click in the "Plain Text" checkbox.

> Note: When you select the Plain Text checkbox, your message is sent as-is, even if it contains some HTML markup. KnowledgeSync will not parse the message to add any headers or metadata that might be required for the message to render.

Select one of the following actions (if applicable):

- Include Reports as Email attachments
- Include Files as Email Attachments
- Email Notifications Wait until Reports Are Completed

For more information, see E-Mail Attachment Handling.

Please note that KnowledgeSync's Email server must be enabled to process if you will be sending email alerts. To view the status of (or start) the Email server, click on the Monitor menu option, select the "Activity" option and locate the "Email Server". The status of this server is shown and you and may toggle the server "On" or "Off".

Note that if you go to the "Email Delivery" branch of the Monitor, you may view email alert messages that have been sent and are waiting to be sent.

5.12.9 Plain Text Email Design
The following is an example of a plain text email alert message:

Dear {Contact_Name},

The following items have shipped against your purchase order {Customer_PO}.
Part Number   {Part_Number}
Part Description {Description}
Shipped Date   {Packlist_Date}

THE EMAIL ALERT MESSAGE IN PLAIN TEXT FORMAT

You may select database fields for your message by positioning your cursor where you want the field to appear within your alert message and then double-click on the desired field from the list of "Available values".

5.12.10 HTML Message Design

The HTML editor is available from the Email and Webcast delivery methods and is available (for email) only if you are using an Internet/SMTP mail sending account. You create your HTML content by clicking on the "Preview Message" tab and then directly entering your desired HTML content (and formatting) from within this tab.

If you prefer to directly enter (or cut-and-paste) HTML <u>syntax</u> for your alert message, you can click on the "Raw Message" tab and do so – and KS will automatically insert the commands {BEGIN*HTML} and {END*HTML} at the beginning and end of your raw HTML syntax once you click off of (and back onto) the "Raw Message" HTML option.

In addition to more advanced formatting (fonts, colors, etc.), the HTML editor is often used to create alerts that contain tables – an option not possible with plain text. One of the most useful HTML formatting tools is the "create table" button:

With this button you can insert a table of 'x' rows and 'y' columns, and then insert the corresponding column headers and data field names as shown in the following illustration:
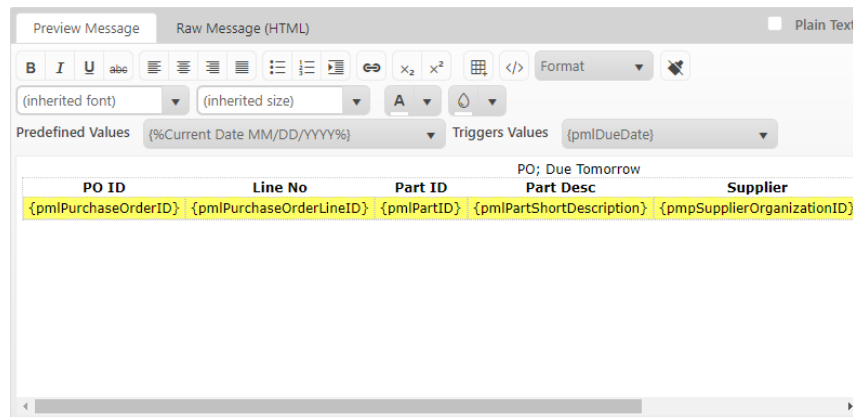
EMAIL ALERT: AN EXAMPLE OF AN HTML TABLE CONFIGURATION

(The green section indicates usage of the "begin and end repeat" functionality discussed in the following section.)

5.13 Begin & End Repeat (Aggregate Vs Non-Aggregate Events)

Every KS event is either an aggregate event or a non-aggregate event; if an event includes the commands {BEGIN*REPEAT} and {END*REPEAT}, it is an aggregate event. If an event does not make use of these commands, it is a non-aggregate event.

In a non-aggregate event, each <u>record</u> that meets the event's criteria triggers an alert (via email, text, and/or webcast), each <u>record</u> triggers the generation of a report or chart associated with the event, and each <u>record</u> triggers the execution of any actions associated with the event.

In an aggregate event, <u>all records</u> are put into one alert; they <u>collectively</u> trigger a single generation of a report or chart associated with the event, they <u>collectively</u> trigger a single execution of any actions associated with the event.

Please review all of the following sections on the use of the {BEGIN*REPEAT} and {END*REPEAT} commands.

The commands {BEGIN*REPEAT} and {END*REPEAT} (located at the bottom of the list of date variables "available values for use in message content") control whether an event "repeats" the details of multiple records within a single alert message. For example:

An event that generates one alert message for each record that meets the event's criteria does <u>not</u> use {BEGIN*REPEAT} and {END*REPEAT}.

An event that puts the details of multiple records that meet the event's criteria into a single alert message uses {BEGIN*REPEAT} and {END*REPEAT}.

If you have a report associated with an event that uses {BEGIN*REPEAT} and {END*REPEAT}, the report will generate once per alert message that is sent. Similarly, if you have an action associated with an event that uses {BEGIN*REPEAT} and {END*REPEAT}, the action will execute once per alert message that is sent.

If you do not use {BEGIN*REPEAT} and {END*REPEAT} in an event, the event will execute its alerts, its reports, and its actions once for each record that matches the event's criteria.

The {BEGIN*REPEAT} and {END*REPEAT} commands can appear in one or more of the following tabs within an event: Email, SMS Text, Webcast, or Create File.

> Note that an individual event may send either one alert per triggered record, or one alert per group of triggered records; an event may NOT have some delivery methods send one alert per triggered record and other delivery methods send one alert per group of records. (You would need to configure two separate events to do this.)

Important:

Be careful when using {BEGIN*REPEAT} and {END*REPEAT} in an event, as it must not be used if an event needs to trigger its alerts, reports, or actions once for each record that matches the event's criteria.

5.13.1 Repeating Text in an Alert Message

{BEGIN*REPEAT} and {END*REPEAT} let you configure an alert message to contain the details of multiple triggered records. Consider the following alert message configuration:

```
The following order was shipped later than the customer's required date:

Order ID:          {OrderID}
Company:           {CompanyName}
Required Date:     {RequiredDate}
Shipped Date:      {ShippedDate}
Salesperson:       {FirstName} {LastName}
```

If you leave the message like this, KnowledgeSync will send out one alert message per triggered record (i.e., one alert per late-shipping order). But if you wanted to send out one alert message with the details of all matching records (all late-shipping orders) in it, you would use the "repeat" commands as follows:

```
The following orders were shipped later than the customer's required date:

    {BEGIN*REPEAT}
    Order ID:          {OrderID}
    Company:           {CompanyName}
    Required Date:     {RequiredDate}
    Shipped Date:      {ShippedDate}
    Salesperson:       {FirstName} {LastName}
    {END*REPEAT}
```

Advanced uses of the {BEGIN*REPEAT} and {END*REPEAT} commands are detailed later in this chapter.

5.13.2 Using an HTML Table

If you are creating an HTML alert that includes a table, your HTML alert text will resemble the following (very basic) example:
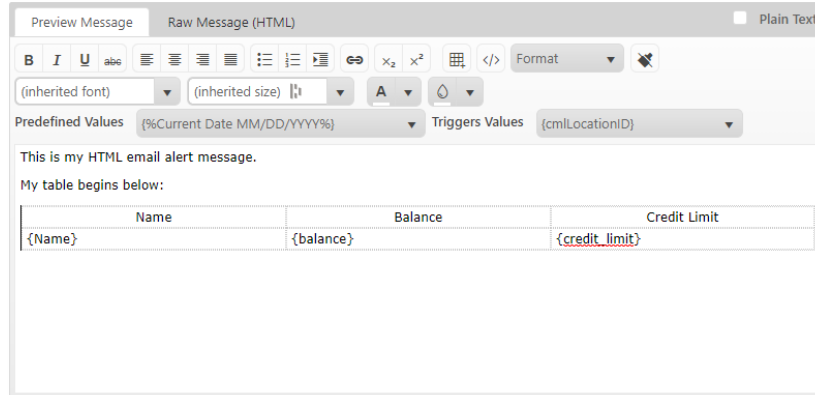
HTML TABLE

The "raw" HTML syntax will look like this:
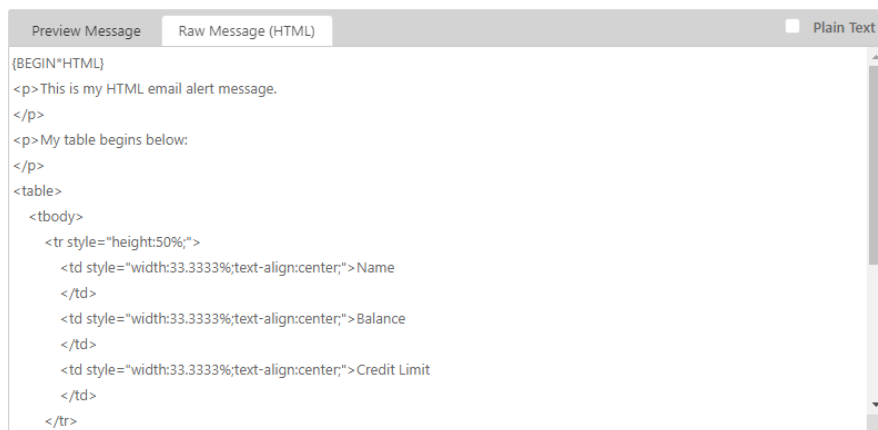
RAW HTML SYNTAX

5.13.3 Repeating Records in an HTML Table

The {BEGIN*REPEAT} command needs to go directly after the table's column headers, and before the table's detail lines:
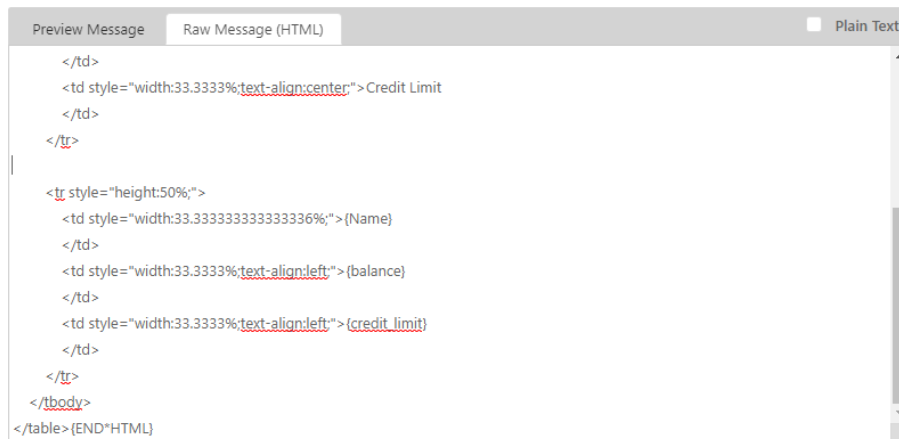
INSERT A BLANK LINE TO ACCOMMODATE {BEGIN*REPEAT}

When you double-click on the {BEGIN*REPEAT} command, the following line will be inserted into your HTML:

{BEGIN*REPEAT} INSERTED INTO HTML TABLE

You'll notice that the {BEGIN*REPEAT} command is accompanied by some additional HTML syntax; please leave this syntax as is.

Your next step is to insert the {END*REPEAT} command; begin my inserting a blank line after all of the data fields for the table's rows have been specified:

INSERT A BLANK LINE TO ACCOMMODATE {END*REPEAT}

When you double-click on the {END*REPEAT} command, the following line will be inserted into your HTML:

{END*REPEAT} INSERTED INTO HTML TABLE

You'll notice that the {END*REPEAT} command is accompanied by some additional HTML syntax; please leave this syntax as is.

To see what the HTML table – with "repeat" enabled – will look like, click on the "Preview Message" tab and you'll see a display similar to the following:
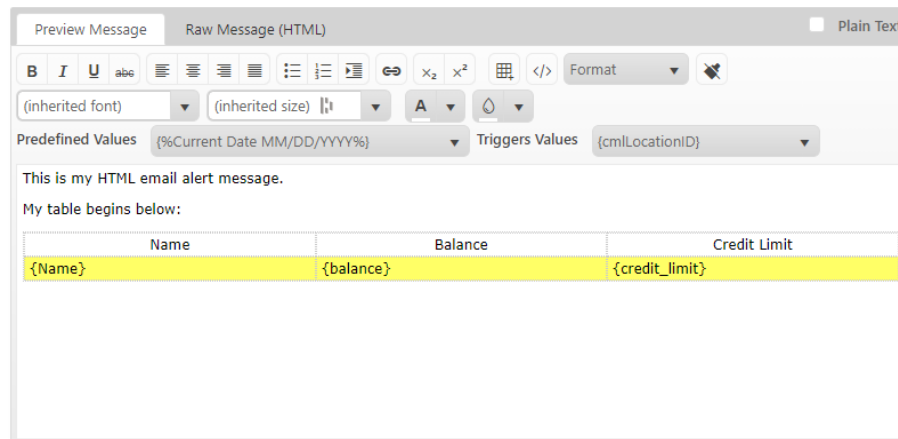
HTML TABLE WITH "REPEAT" SECTION HIGHLIGHTED

5.13.4 Grouping Messages ("Repeat" with a Break Field)

The {BEGIN*REPEAT} and {END*REPEAT} commands allow you to send one message that contains all matching records. There is a second "repeat" option – sending one message per "group" of matching records.

For example – if you had an event that retrieved sales orders, you might want to send your CFO one message for each region in which sales occurred; each message containing a list of all of the sales in a specific region.

This is referred to as "grouping messages" and to do so, you need to do two things: 1) "sort" the event's trigger's matching records so that similar records are grouped together, and, 2) tell KnowledgeSync what field it should use to indicate when it should "break" to a new message. Follow these steps:

1. Edit the event's <u>trigger</u> and make sure the field that you wish to sort on (e.g., the region field) is among the data being retrieved.
2. Sort the trigger according to this field (e.g., sort by sales region).
3. Go to the event's alert message and include {BEGIN*REPEAT} and {END*REPEAT} in the message.
4. Go to the event's "advanced" subscribers and click in the <u>first</u> field in the <u>upper</u> grid (under "Message Break Field"). Click on the list button select your trigger's break field (e.g., sales region). Do not select anything else in the upper grid.
5. When this event runs, the trigger will group matching records by region. Because of the "break field", KS will break to a new alert message whenever the sales region changes. The configuration looks like this:
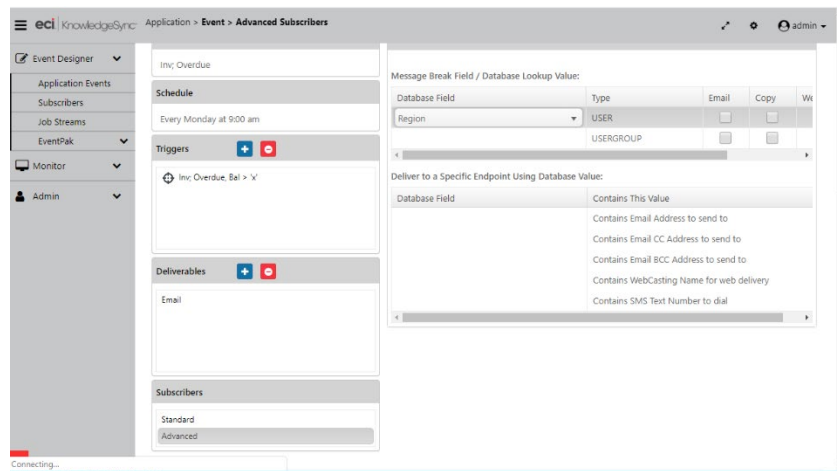
REPEAT ALERT WITH A MESSAGE BREAK FIELD

5.13.5 Grouping Messages by Recipient

One of the most common uses of "grouping" records is to group messages <u>by recipient</u>.

For example – an event that sends overdue invoices to customers. You want to send each customer <u>one message</u> that contains <u>all</u> of their overdue invoices.

When you wish to group records by recipient – <u>and deliver those records to those recipients</u> – you do <u>not</u> use the message break field in the <u>upper</u> part of the Subscribers → Advanced grid. Instead, you use the first field in the <u>lower</u> part of the Subscribers → Advanced grid ("Contains email address to send to").

When an event uses {BEGIN*REPEAT}/{END*REPEAT} and, at the same time, has specified a value in the Subscribers → Advanced grid field titled "Contains email address to send to", <u>KnowledgeSync automatically breaks to a new message when the alert recipient changes</u>.

So – here's what you'd do to configure an event that sends each customer a single email alert containing all of their overdue invoices:

1. Edit the event's trigger and make sure the field that contains the recipient's delivery address (e.g., the customer's email address) is among the data being retrieved by the trigger.
2. Sort the trigger according to the recipient's delivery address (e.g., the customer email address).
3. Include {BEGIN*REPEAT} and {END*REPEAT} in the alert message.
4. Go to the event's "Subscribers → Advanced" option. Click in the <u>first</u> field in the <u>lower</u> grid and click on the list button that appears there ("Database Field").
5. Since you want to deliver the alert to the email address of the customer who has overdue invoices, select the data field that contains the customer's email address.

This event could send alerts to <u>both</u> the customer and their salesrep. You would make sure that your trigger also retrieves the email address of the customer's salesrep, and then you'd select the salesrep's email address field as the field that "Contains the Email CC Address to send to"
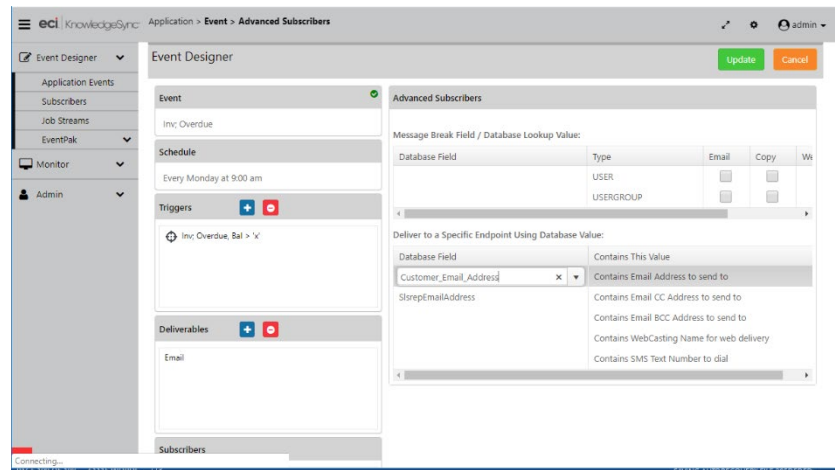
SENDING AN ALERT TO "RELATED" PARTIES
(THE CUSTOMER & THE SALESPERSON)

5.13.6 Grouping Messages Per Recipient AND Break Field
Consider the following scenario:

You want to send each salesrep a list of their sales per region – one alert <u>for each region</u> in which they have closed sales. To configure this, you would:

- Go to the event's trigger and make sure the field that contains the recipient's delivery address (e.g., the salesrep's email address) is being retrieved.
- Make sure that the "break field" (e.g., the sales region) is also retrieved.
- Sort the trigger's matching records first by the recipient (the salesrep's email address), and second by the event's break field (the sales region).
- Include {BEGIN*REPEAT} and {END*REPEAT} in the alert message.
- Go to the "Subscribers → Advanced" option in the event.
- Click in the <u>first</u> field in the <u>lower</u> grid and select the recipient's delivery address (the salesrep's email address).
- Click in the <u>first</u> field in the <u>upper</u> grid and select your break field (the sales region).

When this event runs, the trigger will group matching records by recipient (salesrep). Because of the "break field" specified in the upper grid, KnowledgeSync will break to a new alert message whenever the value of the break field (the sales region) changes. Each salesrep will get a separate email message for each region in which they had sales. The configuration looks like this:
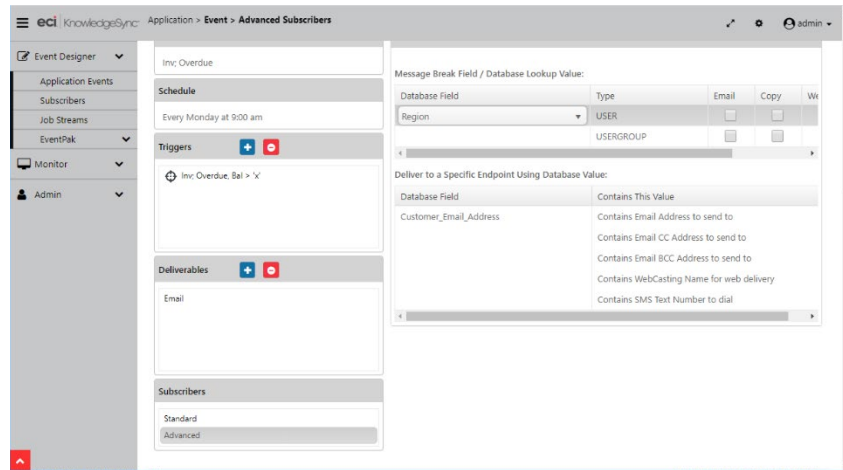
MESSAGE BREAK WITH DYNAMIC RECIPIENT

This kind of configuration is very typical in sales order related events (order confirmation, shipment confirmation, etc.), where an order consists of multiple lines. You need KnowledgeSync to create a separate message for each order (and email it to the corresponding customer and/or salesrep). Thus the lower Advanced Subscriber grid would refer to the fields containing the customer's and salesrep's email addresses.

At the same time, since an order may consist of multiple lines, you need KnowledgeSync to put all the lines for an order in a single email – and break to a new email only when there are no more lines for a given order.

Thus the upper Advanced Subscriber grid would refer to the "order number" field – telling KnowledgeSync to "break to a new message" when a new order is encountered.

5.14 Advanced Message Formatting Options
The following sections cover additional formatting controls for alert messages.

5.14.1 Controlling the Display Length of a Text Field
You can control the display length of a text field in an alert message. To specify the maximum display length, add a colon followed by the maximum display length, followed by the plus sign ('+') as in:

    {Item_Description:25+}

Depending on the email font that an alert recipient uses, this function may also be used to line fields up in a columnar format, as in:

    {Item_Description:20+} {Item_Type:6+} {Item_Class:10+}

This method will work if an email recipient is not using a proportional spacing font; if they are using proportional spacing, it is suggested that you use an HTML table instead.

5.14.2 Currency Formats
Currency values are formatted according to the regional currency settings for the server on which KnowledgeSync is installed. If a currency field's length is shortened, KnowledgeSync will allocate the indicated number of characters when the field is placed in an alert message.

{sales_amount$}	$1250.00 (right-justified)
{sales_amount:10$}	$1250.00 (right-justified; allocate 10 spaces)
{sales_amount&}	$1250.00 (left-justified)
{sales_amount:10&}	$1250.00 (left-justified; allocate 10 spaces)

5.14.3 Numeric Formats

Numeric values are formatted according to the settings for the server on which KnowledgeSync is installed. If a numeric field's length is shortened, KnowledgeSync will allocate the indicated number of characters when the field is placed in an alert message.

{sales_units#}	85.50	(right-justified; 2 decimal places)
{sales_units:10#}	85.50	(right-justified; 2 decimals; 10 digits)
{sales_units%}	85.50	(left-justified; 2 decimal places)
{sales_units:10%}	85.50	(left-justified; 2 decimals 10 digits)
{sales_units!}	85	(right-justified; no decimal places)
{sales_units:10!}	85	(right-justified; no decimal; 10 digits)
{sales_units*}	85	(left-justified; no decimal places)
{sales_units:10*}	85	(left-justified; no decimal; 10 digits)

5.14.4 Skipping Fields With Null Values

When formatting an alert message, you may wish to "skip" (allocate no space) to a field if its value is null. (This function checks for "null" only; not for blanks.)

Here's an example:

```
{ItemNo:15+~}{NonInvNo:15+~}{MiscChgCode:15+~}{CommentCode:15+~}
```

The tilde (~) indicates that if a field's value is null, no space will be allocated to it. This tilde must be the final formatting command in a field.

These options are available inEvent Designer that recognize a null or an empty value when generating a deliverable:

- Empty or null to specific value (new)
- Specific value to empty or null (new)
- Specific value to Other value (existing)

5.14.5 Including Both HTML & Plain Text

You can create a single e-mail message that contains both HTML and plain text. For example:

The following customer had more than 2 shipments arrive late last quarter:

```
Company: {CompanyName}
Number of Late Orders:{Late_Order_Count}
{BEGIN*HTML}
Your HTML content would go here
{END*HTML}
```

If the recipient's email system does not support HTML, they will see only the top section of the message; the content of the HTML will appear as an attachment. If the recipient's email system does support HTML, they will see only what is between the {BEGIN*HTML} and {END*HTML} labels.

5.14.6 Embedding URL Links in Message

You can send an email with a URL link in it (such as a link to track a shipment). Make sure that any information required within the URL link (such as the tracking number) is one of the chosen columns in the underlying trigger.

Then, you could embed a link such as the following (a FedEx example):

http://www.fedex.com/cgi-bin/tracking?action=track&tracknumbers={fedexnumber}

{fedexnumber} is the value from the underlying trigger.

5.14.7 Advanced Formatting – Using a Crystal or SSRS Report

If you need more sophisticated formatting for your alert message (e.g., headers/footers, page breaks, sub-totaling, etc.), or you wish to send a standard form or document (like an invoice, quote, statement, et cetera), it is suggested that you configure your KnowledgeSync event to use a Crystal or SSRS report as the alert content. KnowledgeSync can populate and generate these "report forms" and then deliver them to their intended recipients. (More on this in the chapter on "Reporting".)

5.15 E-Mail Attachment Handling

- **Include Reports as email attachments**. If an event has associated reports, you may specify whether the reports will be attached to the outgoing email. If you will be using KnowledgeSync to generate and distribute a large number of reports, it's more efficient to <u>distribute</u> the reports via FTP/SFTP or webcast and use email to <u>notify</u> people when those reports are available.
- **Include Files as email attachments**. If an event has associated files, you may specify whether the files will be attached to the outgoing email. If an event's trigger includes an object called "attachment_files", the files referenced in that object will also be attached.
- **Email notifications should wait until reports are completed**. If an event has associated reports which are <u>not</u> going to be attached, you should have the alert <u>wait</u> to be sent until the reports are finished generating.

5.16 What Needs to Happen to Send E-Mail Alerts

In order to send email alerts, <u>all</u> of the following need to be true:

- The event must have an email subject, message text, and a valid email sending account.
- There must be at least one subscriber (standard or advanced) selected to receive alerts via email who is "active" with valid email addresses, and their "delivery times" must allow message receipt.
- The Windows or Desktop service and the email server must be running.
- The event's trigger must retrieve matching records and if the event has been triggered (and no new matching records exist), you must either remove triggered items or the event must "repeat notifications for triggered items".

5.17 SMS Text Messaging

A subscription to the Twilio text messaging service is required to utilize this option.

SMS texting generally follows the same rules as configuring a plain text email alert – except that due to the brevity of a typical text message, you will most likely not be using the begin and end "repeat" commands.

In addition to specifying the contents of the text message, be sure to specify the following:

The name of the SMS account used to send the text. This is configured in the KnowledgeSync Administrator.
The phone number that the text will be sent from. Note that Twilio requires phone numbers to be registered before you can send texts from them. For trial accounts, Twilio assigns a single phone number that you can send from. With a paid subscription, you can register your own numbers and send from any of them.

This number <u>cannot</u> come from the event's trigger, it has to be one of the numbers in the dropdown – KnowledgeSync connects with Twilio to retrieve the numbers associated with the specified account.
The contents of the message (which can be any combination of hard-coded text, date variables, or data variables from the underlying triggers)
(If the event has any associated reports or charts) Whether the text message should wait for those items to complete their generation before the text message is delivered.

5.18 Webcast (Dashboard Alerts)

The webcasting ("dashboard alerts") module requires a few additional steps for configuration; please refer to the chapter titled "Dashboard Alerts" for details.

5.19 Charts & Reports

Charts and/or reports are associated with events for three reasons:

- To generate a form (such as an invoice or statement; designed with either Crystal Reports or Microsoft SQL Server Reporting Services) that is to be delivered as part of the alert.
- To generate an analytical report (Crystal or SSRS) associated with the event being triggered.
- To generate a chart or graph (designed within KnowledgeSync) that contains useful information.

Please refer to the chapters on "Reports" and "Charts" for more details on these options.

5.20 Files

You may associate static files to an event, such as MS Word or MS Excel files. Associated files may be delivered via email, FTP/SFTP, or webcast. Potential uses of static attached files include:

- To send a "Policies and Procedures" document along with a message
- To attach a product brochure to an email going to a prospect
- To send an HTML newsletter to recipients
- To send an MS Excel spreadsheet to specific recipients

If you wish to add a static file to an event, go to the event's "Deliverables", click on the blue "Plus" sign, and under "Alert Content", select the option for "File".

Copy and paste a file path from your File Explorer of the file you wish to add to the event.

If you are running the KnowledgeSync Windows service and web portal app on different machines, you will need to make sure that any such files are in a network path that is accessible from both machines

Once you have specified the file path, click on the "Add" button.

You have the following additional options when associating files:

Include Files in Copy operations. Whether files selected for this option will be included for any subscriber for whom the "Copy" (FTP/SFTP) delivery method has been selected. (Do not refer to mapped drives in your file specifications; use UNC format instead.)
File Description. An expanded description of the file's contents.
Ascii. Whether the file's contents are in ascii format (check 'on' for ascii format).

Your selected file(s) will appear as follows:
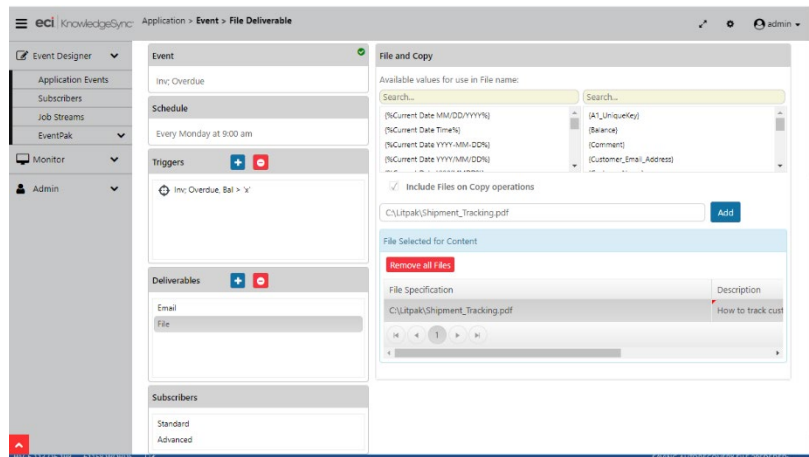
AN EVENT'S STATIC FILES

5.20.1 File "Alternative Text" Option

This option is available only if sending email via Internet/SMTP method.

KnowledgeSync gives you an easy way to deliver HTML documents without having to specify their contents in the Email component of an event. This uses the "alternative text" option from within the "File" deliverable.

Once you have selected a file for an event, click in the "Description" field of that file and click on the list button that appears there. You may choose from three "alternative text/html" options (HTML, richtext, enriched). When you do so, the associated file will appear within the underline body of the outgoing email message sent by an event (as opposed to being delivered as an attachment).
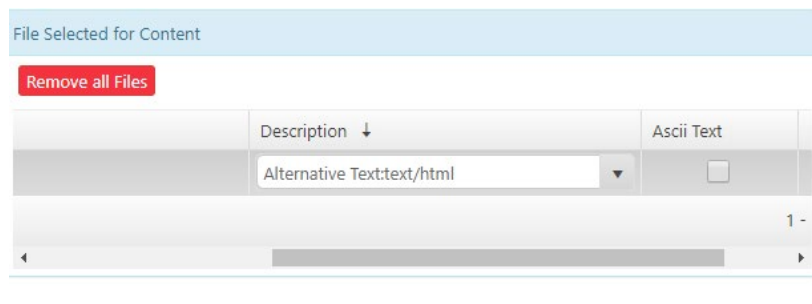
EVENT FILE "ALTERNATIVE TEXT" OPTIONS

If an event delivers content to some users whose mail system can display HTML and other users whose mail system cannot display HTML, you can both associate an HTML document as "alternative text", and also specify the same text (in plain text format) within the email deliverable of an event.

No trigger data fields may be included within the alternative text file option and the multiple "alternate" format options allow you to have multiple alternative files of various formats for the same event.

5.20.2 Actions (Workflow) Deliverables

The last group of an event's Deliverables are its "Actions":
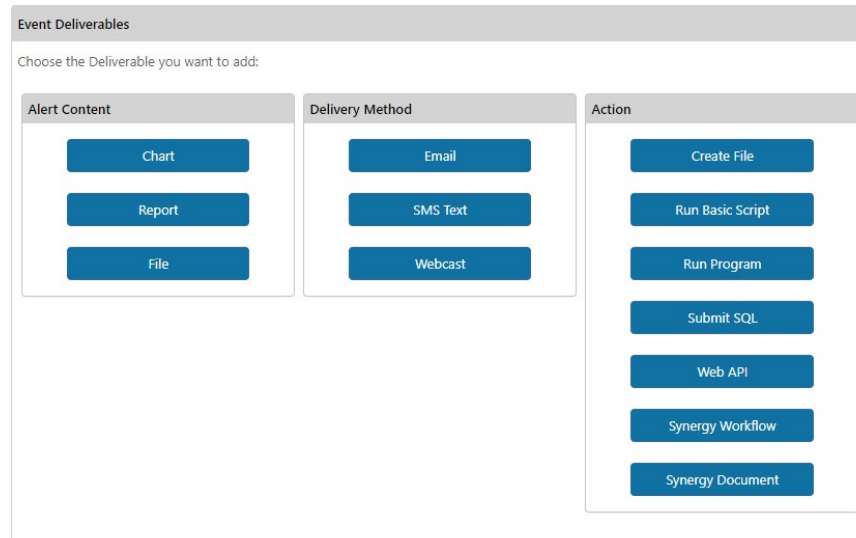
EVENT "ACTIONS" (RIGHT-HAND COLUMN)

Remember: If an event needs to <u>execute one action per record</u> that meets the event's criteria, the event must <u>not</u> use {BEGIN*REPEAT} and {END*REPEAT} anywhere in the event's configuration.

5.20.3 The Seven Types of Workflow Actions
An event may execute any combination of the following actions:

- Create a file containing content from a triggered event.
- Run VB (visual basic) scripts.
- Run executable programs.
- Execute SQL (structured query language) statements
- Make REST web API calls.
- Create a workflow request in the Exact Synergy application (not available in all versions of KS)
- Create a document in the Exact Synergy application (not available in all versions of KS)

All of these may incorporate data from the corresponding triggered event and the order in which they are executed is the order in which they appear in the window.

If you have an event which needs to execute an action <u>once per triggered record</u>, and also needs to generated a <u>summarized alert message</u> (using {BEGIN*REPEAT} and {END*REPEAT}), you'll need to configure <u>two events</u> – one to do the summarized alert, and another to do the workflow action.

If an event has associated files – either from as "Files", or if the event's trigger uses the "attachment_files" column -- those files will be stored in a variable called "attachment_files" which may be used in your workflow actions.

Except for the "create file" action, if an action is associated with an event, it (the action) typically needs to <u>run once for each record returned by the event</u>. In such a case, the event to which the action is associated must <u>not</u> use the {BEGIN*REPEAT} and {END*REPEAT} commands anywhere in its configuration.

5.20.3.1 Create File
An event can take its "triggered" data and write it out to a file as in the following example:
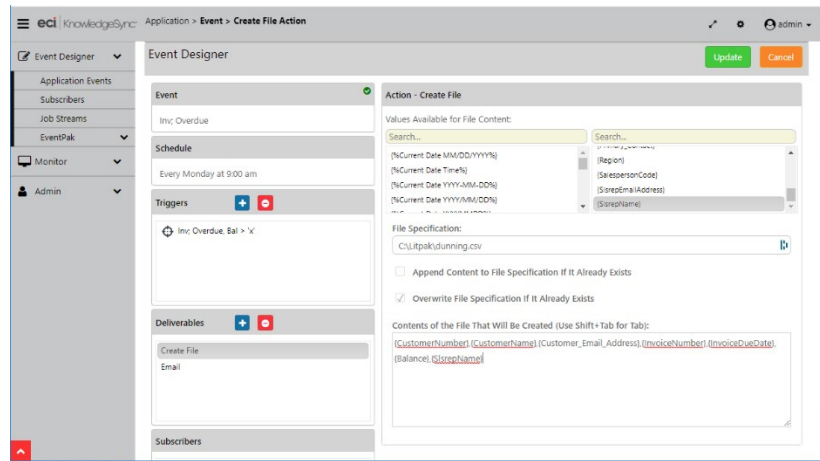
WORKFLOW ACTION: "CREATE FILE"

Files created in this manner can be used for import into another application, to create log files of triggered event data, or to create a file for a mailing or fulfillment house. The file can contain hard-coded text, triggered field values, and date variables. For each file you need to specify:

File Specification. The name of the file to be created. This name may be hard-coded or it may include field and date variables. Do not reference mapped drives.
Append content to file. If the named file already exists, checking this box will cause the event to add its data to the existing file.
Overwrite file specification. If the named file exists, checking this box will cause the event to delete the existing file and overwrite it with new contents.

Click in the box beneath "Contents of the file" and begin entering any combination of hard-coded text and field (and date) variables. You may separate fields by using a comma, semi-colon, or any other valid punctuation. Use Shift/Tab to create tab-based separations. If the event's trigger uses the {attachment_files} function, you may reference this field as a data field variable.

You can use {BEGIN*REPEAT} and {END*REPEAT} in your file as well as the field formatting commands (detailed earlier in this chapter).

5.20.3.2 Run Basic Script
The second action that an event can execute is to run a Visual Basic ("VB") script.

Before you can select a script for an event, the script must be defined within KnowledgeSync; you can define scripts by expanding an application's events and selecting the option called "Basic Scripts". From here you can view, create (as well as copy & paste), modify, and delete scripts. Once you have created VB scripts, you may select them for use by an event – when the event's conditions are met.

All of the data returned by an event's triggers can be used by the script.

To associate a script with an event, go to the Deliverables for an event and click on the blue "Plus" sign. Select the action called "Run Basic Script" and select the desired script.
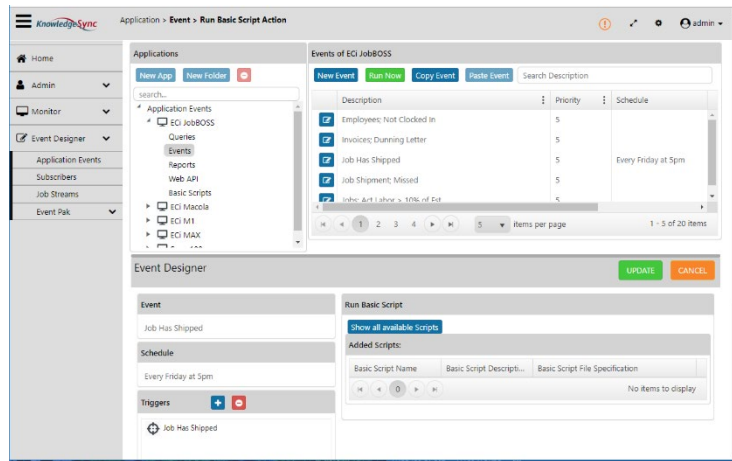
"RUN BASIC SCRIPT"

The use of VB scripts in KnowledgeSync require expertise in Visual Basic programming syntax. Sample scripts are available from KnowledgeSync's Support department by submitting a request to:

http://www.KnowledgeSync.com/support_email.php

KnowledgeSync Support Services does not include support of custom VB scripts. Quotations for fee-based consulting services for the creation and/or implementation of VB scripts are available on request.

You can display a list of currently-defined VB scripts, select one or more desired scripts and use the "Add" button to add them to your event.

As all KnowledgeSync events run in an automated manner, please be sure that all scripts used in KnowledgeSync do <u>not</u> prompt for any user input.

5.20.3.3   Run Program
The option to configure an event to "Run a Program" when an event's conditions are met looks like the following:
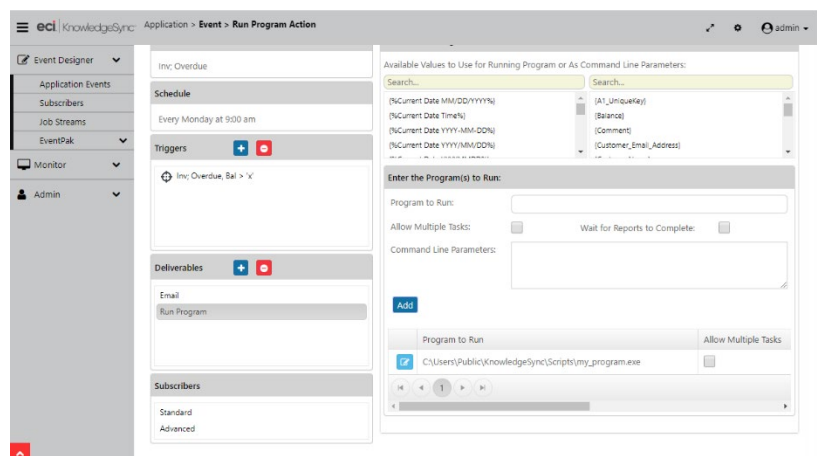
"RUN PROGRAM"

Program to run. Type in (or copy and paste) the name of the executable program you wish the event to run. The file specification must not use mapped drives.

Allow Multiple Tasks. This lets you specify whether the event will run only a single copy of this task at any one time, or allows multiple copies to run simultaneously. (The suggested setting is <u>unchecked</u>, to optimize system resources.)

Wait for Reports to Complete. If the corresponding event has associated reports, you may wish for the reports to generate before your program is executed. (E.g., you may wish to run a report on data <u>before</u> it is updated by a program associated with this event.)

Command Line Parameters. Events can make use of command line parameters to pass data from triggered records into an executable program. Parameters may include hard-coded values and data field (& date) variables; be sure to enclose all date and text-based data field variables within single quotes as in '{ACCOUNT_ID}' (not necessary with numeric fields). Data fields may also use any of the formatting controls detailed earlier in this chapter.

If an event trigger uses the {attachment_files} function, you may also use this as a field variable.

To add parameters, click in the "Command Line Parameters" field and then drag and drop the values you wish to use from the "Available values" lists. Once you have specified any parameters, click on the blue "Add" button to add the program to the current event.

> *Any program an event runs must not prompt for user input. And, if a program calls any "dll" files, be sure to put those dll files in the KnowledgeSync Windows Service machine's "windows/system" directory.*

5.20.3.4 Submit SQL / Stored Procedure
SQL statements (& stored procedures) can be used to add, update, or delete records from an application database. Note the following:

If performing a database update, your SQL must include a "where clause" to ensure that it updates only the appropriate records. (There is no implied where clause.)
All standard SQL syntax (for the database & driver selected) is supported

To add one or more SQL actions (or stored procedures) to an event, go to the Deliverables for an event, click on the blue "Plus" sign, select the option for "Submit SQL", and click on the blue "Add Row" button. You'll see a window like the following:
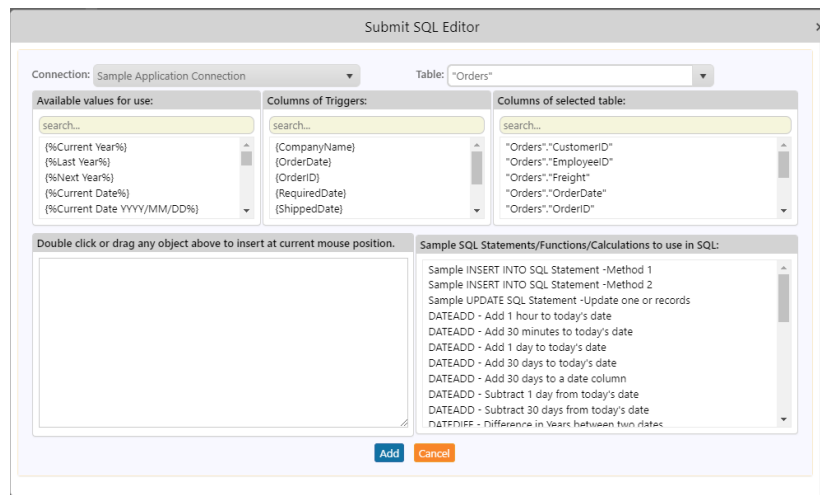
"SUBMIT SQL"

Connection. Click in this field and select the database connection that will be used by the SQL statement or stored procedure.
Table. Click in this field and select the table whose column names you wish to display.
SQL Statement. Enter (either manually type or drag-and-drop) the SQL syntax that will be executed by this event, such as:

```
update sysdba.ACCOUNT set sysdba.ACCOUNT.TYPE = 'Inactive'
```

```
where sysdba.ACCOUNT.ACCOUNTID = '{ACCOUNTID}'
```

You may use any combination of SQL syntax and data field (& date) variables in your SQL syntax; if the event's query uses the {attachment_files} function, you may reference this field as a data field variable.

All standard SQL commands (such as "insert", "update" and "delete") are supported in the SQL action tab.

The "where clause" in your SQL tells KnowledgeSync which records to update.

For assistance with your SQL statement, please refer to the window titled "Sample SQL Statements / Functions / Calculations".

Wait for Reports to Complete. Once you have added a SQL statement to execute, you may scroll the window containing your statement to the right in order to access the option whether your SQL statement should "wait for reports to complete".

If the corresponding event has associated reports, you may wish for the reports to generate before your SQL is executed. (E.g., you may wish to run a report on data before that data is subsequently modified by your SQL.)

To run a stored procedure instead of a SQL statement, choose your database connection and then specify the procedure name, as in "dbo.myprocedurename". To pass parameters into your stored procedure, use the syntax:

```
dbo.myprocedurename '{account}','{rep}',{amount}
```

5.20.3.5 Make a REST API Call
Please refer to the REST API chapter for details.

5.20.4 Synergy Workflow Creation / Synergy Document Creation
These two options will be enabled only if you are using the "Exact Synergy" application and wish for the triggering of an event to create either (or both) a Synergy workflow request, and/or a Synergy document. All of the information from the triggered event may be passed into the components of the Workflow or Document being created in Synergy.
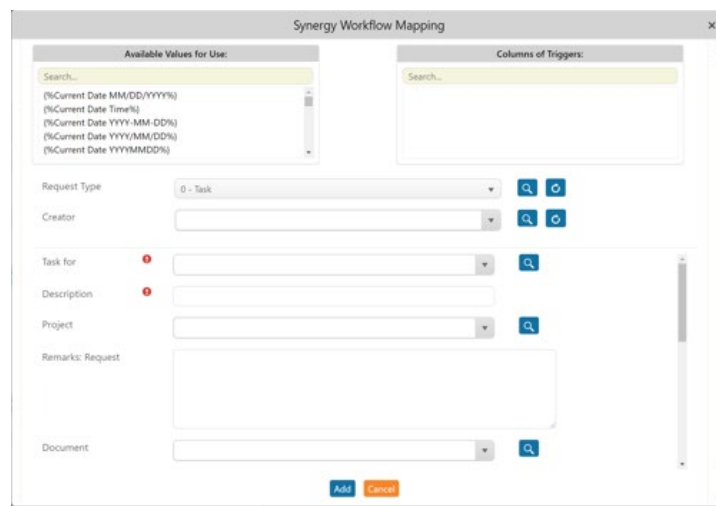
"CREATE SYNERGY WORKFLOW REQUEST"
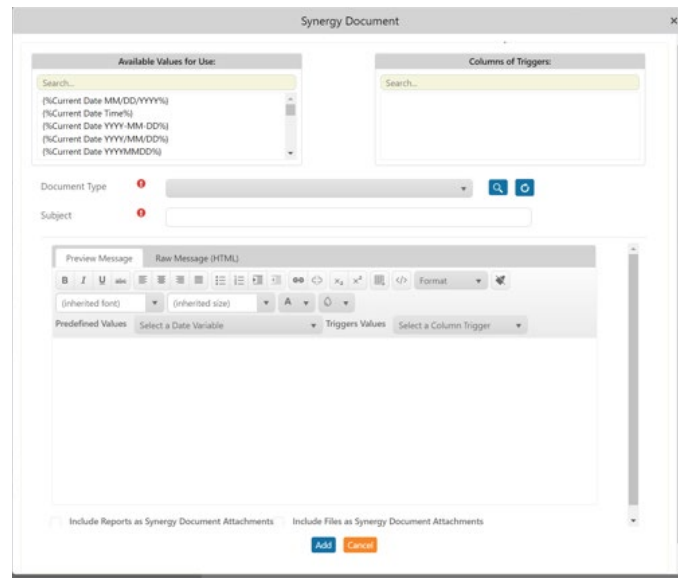
"CREATE SYNERGY DOCUMENT"

Note that the two Synergy options will appear greyed-out if you are not using the Synergy application.

5.20.4.1 Subscribers
"Subscribers" are individual people, groups of people, and computers that will receive alerts, reports, charts, and files that are associated with an event. In addition to specifying who will receive this content, the Subscribers option also lets you specify the method by which this content will be delivered (email, text message, Copy/FTP, or webcast.

To add, view, modify, or remove a subscriber associated with an event, click on the "Standard" or "Advanced" options beneath the "Subscribers" section of the event design window.

The "standard" option is used when you know the names of the groups, individuals, or computers that you wish to delivery event content to.

The "advanced" option is used when you want an event to deliver its content to one or more addresses that are related to the records being triggered – such as delivering an alert to the customer associated with an orders, and/or to the salesperson associated with corresponding customer.

5.20.4.2 Adding/Changing/Removing Standard Subscribers
To add, change, or remove standard subscribers for an event, go to the "Subscribers" section of an event and click on the "Standard" options.

A window like the following will appear:
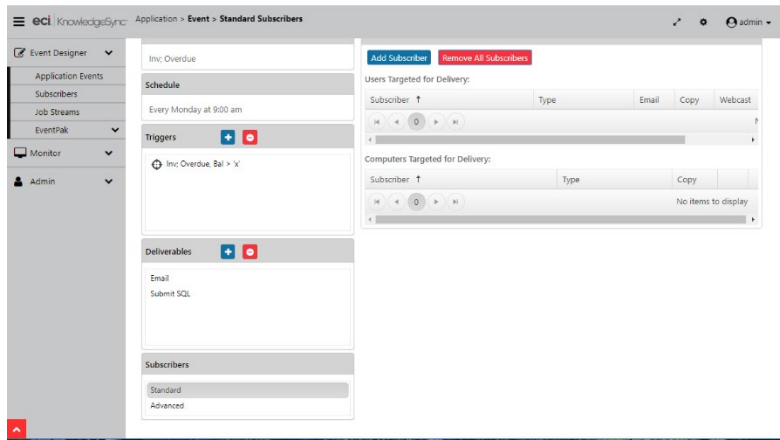
EVENT "STANDARD" SUBSCRIBERS

To add "standard" subscribers (individuals, groups, and/or computers), click on "Add Subscriber" and a window like the following will appear:

SELECTING STANDARD SUBSCRIBERS

Once you place checkmarks next to the desired individuals, groups, and/or computers for an event, click on the blue "Add" button and a window like the following will appear:
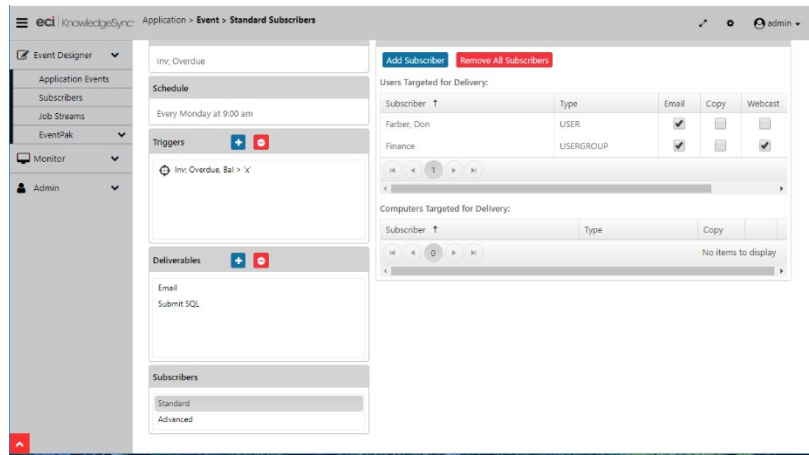
STANDARD SUBSCRIBERS – SELECTED

Place a checkmark beneath the selected content delivery methods for each standard subscriber. Note that the subscribers window scrolls to the right to show the "SMS" (text messaging) alert delivery method.

5.20.4.3 Adding/Changing/Removing Advanced Subscribers

To add, change, or remove advanced subscribers for an event, go to the "Subscribers" section of an event and click on the "Advanced" options.

A window like the following will appear:
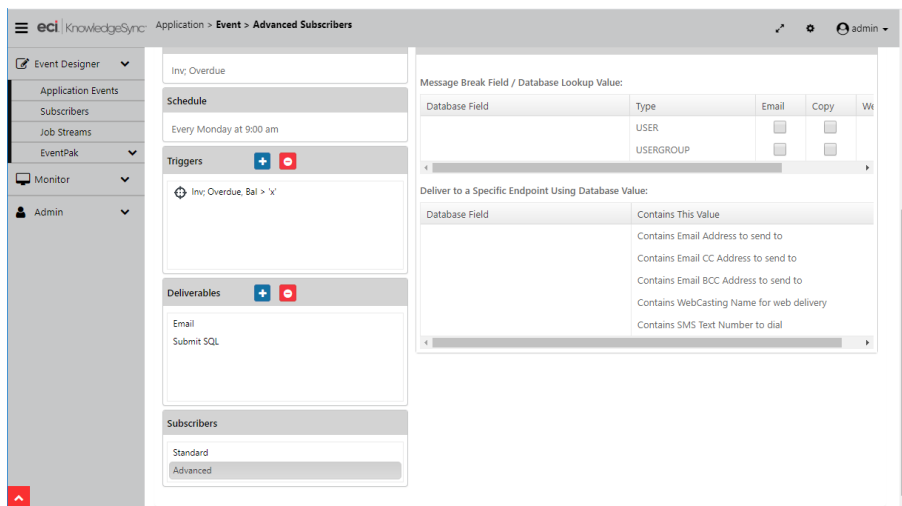
EVENT "ADVANCED" SUBSCRIBERS

The top grid in the "Advanced" tab can be used for two things:

To control the "breaking" of messages that contain the details of multiple records that meet an event's criteria (see section on message breaks earlier in this chapter).
To make use of the "subscriber lookup key" function (detailed in the chapter on "Subscribers").

The lower grid lets you specify that an event will deliver its content to one or more addresses that are associated with the records matching the event's criteria (such a delivering an order confirmation to both the customer who placed the order and the salesperson who is assigned to the customer's account).

Notice that the description of the fields in the lower grid refer to values that are returned by an event's trigger – such as a trigger's value that "contains the email address to send to" – e.g., a query's field that contains a customer's email address.

When you click in any of the fields in the lower grid, a list button will appear, and that button will present a list of values from the triggers associated with this event. Select the values that contain the desired delivery addresses.
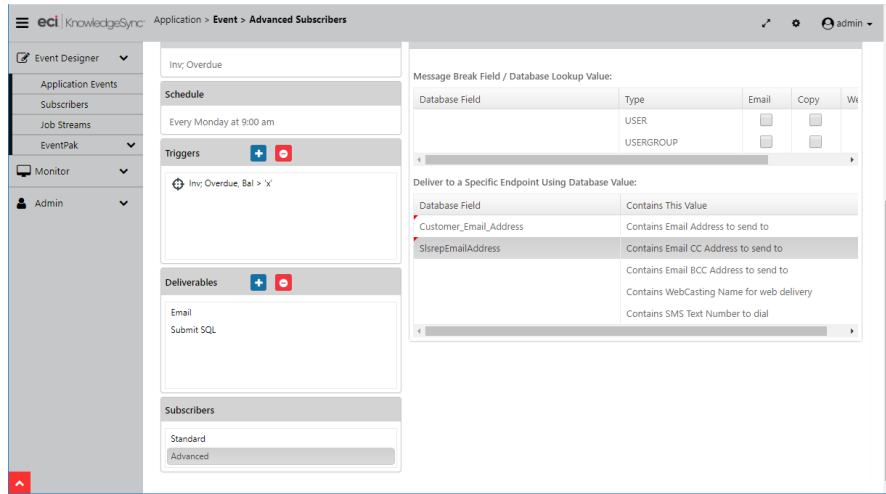
TWO "ADVANCED" SUBSCRIBERS SELECTED

If an event needs to send alerts to <u>multiple</u> addresses for any <u>one</u> of the fields in the lower grid, you'll need to concatenate the address fields (in the event's trigger) into a single database field. Separate multiple email addresses in that concatenated field with a comma.

An event may contain a combination of both standard and advanced subscribers.

5.20.5 Monitoring the Status of an Event's Deliverables

The KnowledgeSync Monitor tracks the status of all deliverables for an event, including whether those deliverables are "pending", have been "completed", and those that are in an "error" state.

If any of an event's deliverables fail to complete successfully because of errors, you may try to correct the error, mark the action as "complete", or manually delete the erroring deliverable. (The Monitor allows for the use of "corrective actions".) KnowledgeSync will attempt to re-deliver an erroring deliverable 'x' times (determined by you).

To see the deliverables for an event, follow these steps:

Open the Monitor, click on the "Application Events" branch and then click on the "Triggered" tab.
Single-click on the blue plus sign next to the triggered event whose deliverables you wish to track.
View the "Deliverables" (alerts, reports, actions) and the event's email alert message (if such a message has been configured for the event).

5.20.6 Job Streams

A job stream is a series of events that need to run in a sequential fashion, one after another. Job streams have the following unique characteristics:

There is a single schedule for all events in a stream. That schedule indicates when the <u>first</u> job in the stream will be submitted.
Each event in a stream **completes all of its components** before the next event in the stream begins.
The "completion" of an event in a stream is defined as its successful <u>execution</u> -- <u>regardless of whether the event is triggered or not</u>.

This last point is <u>very important</u>; if a job stream has 3 events within it, all 3 events will always execute whenever the stream is run. Whether any one of those events is *triggered* is based on the triggers associated with the event.

Job streams are very useful in situations where you need precise control over the order of execution of multiple events; for example, when you need to have an event execute an action to update an application and then wish to have a report run on that updated data. (Two events; one job stream)

> *Regardless of how many components an event has – alerts, reports, and/or workflow actions -- the next job in a stream will not execute until all the components of the preceding event are complete.*

Job streams are valuable when:

You have a series of events that need to be submitted in a specific order, such as an initial event that logs a support ticket from an incoming email, and then a subsequent event that triggers alerts about that ticket.
You have a group of events related to the same business condition, such as a "new hire". Job streams are an excellent way to group related events together, and easily manage them.
You have a business process with decision branching, e.g., a stream that triggers alerts based on either high or low priority support tickets.
In this last example, a stream would contain two events; one event that triggers alerts for 'high' priority tickets, and another event that triggers (different) alerts for 'low' priority tickets. For any one support ticket, <u>both</u> events will always run, but only <u>one</u> event will ever trigger.

The following illustration is an example of how job streams are represented in KnowledgeSync:
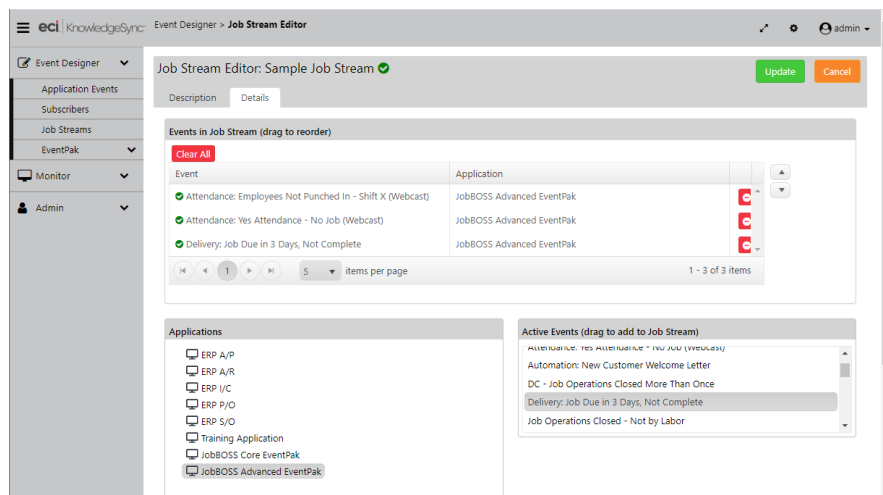
JOB STREAM EXAMPLE

5.20.6.1 Add/Edit a Job Stream
To create a job stream, click on "Job Streams". Use the blue "Add" button to create a new stream, or click on the blue "Edit" menu to view or modify an existing stream.

The following information is associated with a job stream:

Job Stream Name. The name that uniquely identifies the job stream.
Schedule. Click on the List button and choose a schedule that represents when the job stream (the first job in the stream) will execute. It is suggested that the <u>events</u> used within a stream each have their own schedule of <u>none</u>.
Active. Whether the stream is eligible to run.
Notes. A more detailed description of the purpose of the job stream.

5.20.6.2 Selecting Events for a Job Stream

To add an event to a job stream, click on the "Details" tab within job streams.

Under "Applications", select the application that contains the first event you wish to add to the stream; and you will see a list of all <u>active</u> events for that application.
Double-click on (or drag & drop) an event for the stream from the list of "Active Events" to the window titled "Events in Job Stream."
Repeat the preceding step for each event you wish to add to the stream.
Click on "Update" when done.

> *You can drag and drop events in a job stream to re-order them.*

5.20.6.3 Delete a Job Stream

From the initial list of job streams, you can use the red "Minus" sign to delete a job stream from the system.

5.20.7 Show Tracked Items

Within an event's details (once you click on the name of an event in "Edit" mode), you can see whether an event has been checked, has been triggered, and the "unique" IDs of the triggered records.

To see this, click on the option called "Show Tracked Items" as shown in the following illustration:
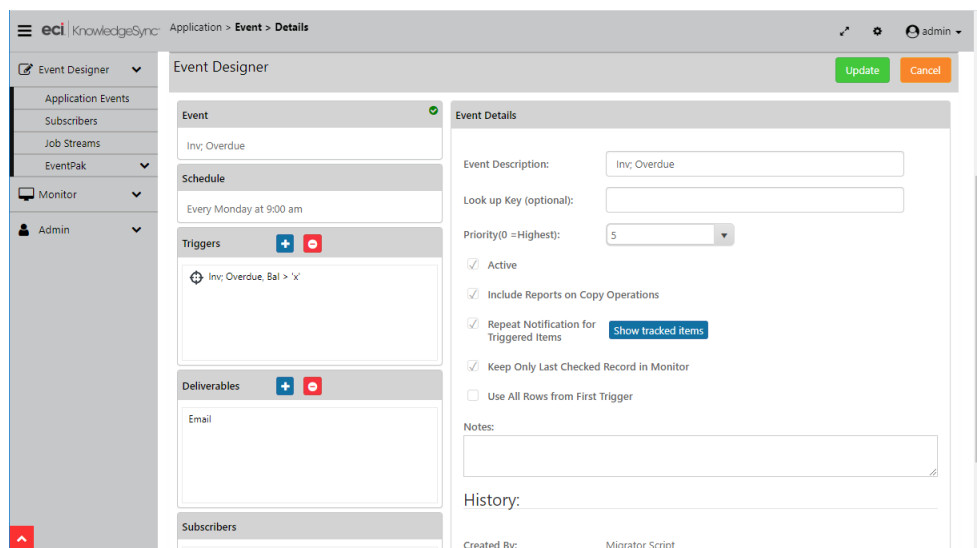
SHOW TRACKED ITEMS BUTTON (MIDDLE-RIGHT)

The resulting window will look like the following:
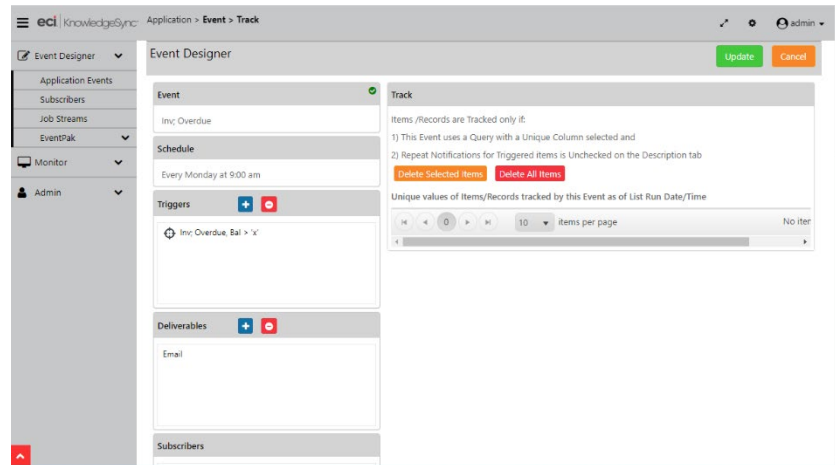
TRACKED ITEMS FOR AN EVENT

From this option you can "un-trigger" (or clear out) the triggered records for an event and thus enable it to be triggered again. You can clear out some or all triggered records for an event.

5.20.7.1 Clearing Triggered Records
To un-trigger a specific record for an event, highlight the record and click on the "Delete Selected Item" buttons. To clear all triggered records for an event, click on the "Delete All Items" button.

5.21 Troubleshooting Events

5.21.1 If Your Event Isn't Executing
Check the Monitor and see if the event appears under "Application Events" that are "pending", "checked", "triggered", or have recorded "errors".

Here are the top reasons why an event may run but not trigger:

- **The event's conditions are not met.** Preview the event's triggers to ensure that there are matching records (if the trigger checks for changed records, the preview will show all records, regardless of a change to their value).
- **The event is already triggered.** Use the "Show Tracked Items" option to see if any triggered records are listed.
- **The email sending account is blank.** Review the event's "Email" deliverable and make sure that the "account to send from" has a valid value.
- **There is no alert message text.** If you've chosen to send alerts via email but have not specified any message text, no alert will be sent out.
- **The subscriber does not have a valid delivery address.** Review the subscriber's profile to see they have a valid address for the chosen method.
- **The subscriber is disabled to receive alerts for that day or time.** Review their profile to see if their "work hours" exclude certain days or times.
- **The corresponding delivery server is disabled.** If an event is configured to send email alerts but the email server is disabled, no alerts will go out.
- **The event is waiting for its report to finish.** If an event has an associated report, it may be configured to "wait" until the report is done before sending its alert.

5.21.2 If Your Event Triggers Over and Over
Check the event trigger's "unique" value. If a trigger doesn't have a unique, the event will trigger repeatedly for the same record, until that record no longer meets the trigger's criteria.

Also check the setting of the event's "Repeat Notification for Triggered Items" checkbox and make sure it is <u>not</u> checked.

5.22 Run an Event Right Now

To run an event once – right now – go to the list of events for a specific application, single-click on the event you wish to submit (to highlight it) and then click on the "Run Now" button. The KnowledgeSync Windows Service must be running to use this option.

6   REST API Triggers

This section documents the features of the REST API trigger.

> *If you are new to interacting with web apps via REST APIs, please review the "KnowledgeSync REST API Introduction" document for a prerequisite look at the concepts needed to work with REST APIs.*
>
> *KnowledgeSync Technical Support assists with the proper use of REST API application features, but they will not assist with troubleshooting individual trigger and action parameters, authentication, JSONpath, and variables.*

6.1   Create a New App

Create a new app as shown in the previous section. If you are only calling a web API, you do NOT need to create a database connection. After creating the Application, click on the "Web API" tree item to display the list of defined APIs. The following window opens if no API definitions have been created yet:
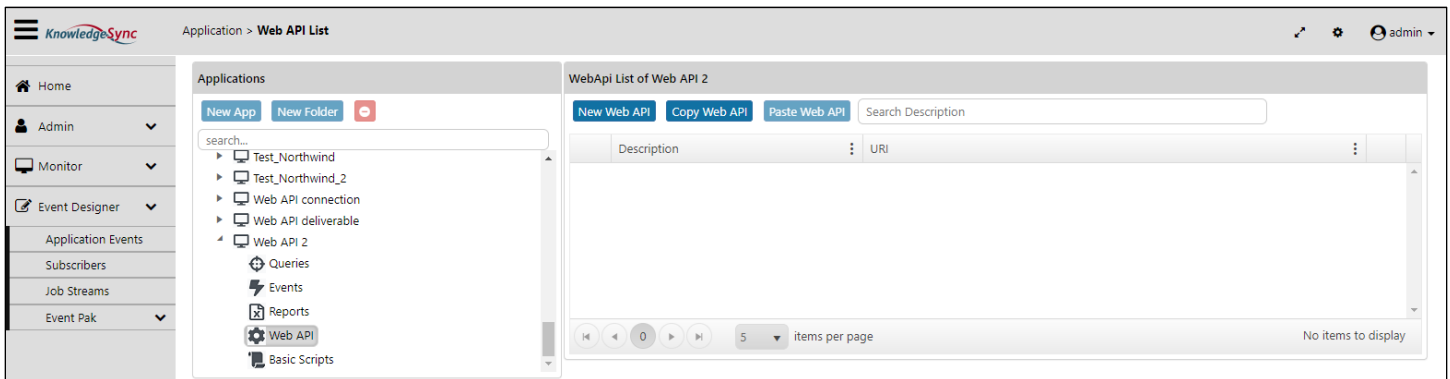
EMPTY WEB API LIST

6.2   Web API Definition

Click the **New Web API** button to display the Web API Designer.

6.2.1   Web API Designer Definition Tab

On the Web API Designer Definition tab, fill in the following values:

- **Description:** The title for your entry, such as "New York Times".
- **Active:** Check this box if you want this trigger to run.
- **Notes:** Any additional notes you want.
- **URI**: The base web address for the API, such as "https://api.nytimes.com/".

KnowledgeSync will replace {bracketed} variables in this field if the variables were defined in a previously-executed trigger, or if it's a standard variable, such as a date variable.

KnowledgeSync will replace {bracketed} variables in this field if you define them using a "URL Segment" parameter as described below.

- **Resource**: The sub-path for the API call, such as "svc/movies/v2/reviews/search.json".

KnowledgeSync will replace {bracketed} variables in this field if the variables were defined in a previously-executed trigger, or if it's a standard variable, such as a date variable.

The URI and Resource fields are appended to each other without a delimiter to create the full URI path. In the examples above, the full URI will become "https://api.nytimes.com/svc/movies/v2/reviews/search.json".

- **Verb**: This is the REST API call verb to use. GET is the default and will be appropriate in most cases.
- **Format**: Select JSON or XML. The parsed XML response and the structure of the XML data are displayed on the Response tab.
- **Ignore Response Code**: Check this box to ignore incoming HTTP response codes from the Web API.

*Note: KnowledgeSync is coded to only accept 201,202 or 204 HTTP response codes. If any other code is sent, KS throws an **InvalidWebAPIResponseException**. This prevents users from being able to use Web API's that use path parameters as part of the URL; for example, a single "404 not found" message breaks the event. By checking this new box, KS will ignore the incoming response code.*

- **API Uses JSON Template:** Check this box if the data parameters in the Web API are defined in a JSON template. KS generates a complex JSON object using the template. (Currently, this is only available for Web API triggers.) See next section for more details.

6.2.1.1   Web API Calls with Advanced JSON

By default, the API Uses JSON Template box is not checked if the following is true:

- For new/existing Web API definitions
- For copied/pasted Web API definitions
- If an eventpak does not contain a value for this flag
- If format is in XML

If the Web API trigger has **API Uses JSON Template** checked:

- JSON is created for every parameter that has a template, using response data from previous query or Web API triggers.
- After the Web API is triggered, all response data from previous triggers is cleared.
- A new response data set is created with the results from the Web API.

The following examples use this data table:

Cust ID	Cust Name	Column C	Contact Code	Contact Name	Column F	Email Address
CUSTA	Acme Corporation	Info C	CONTA	Alison Acme	Info F	aacme@acme.com
CUSTA	Acme Corporation	Info c	CONTB	Bob Acme	Info F	bacme@acme.com
CUSTA	Acme Corporation	Info C	CONTC	Cindy	Info F	cacme@acme.com
CUSTB	Other Corporation	Info CC	CONTF	Roy	Info Y	othera@other.com
CUSTB	Other Corporation	Infor CC	CONTT	Karol	Info R	otherb@other.com

6.2.1.1.1   Advanced JSON Example 1

If the JSON template has a top-level collection (e.g., square brackets [] surround the entire JSON code), a single call is made to the Web API and the JSON collection is passed as a parameter:

```
[{
    "Customer_ID": "{Customer ID}",
    "Name": "{Customer Name}",
    "Contacts":   [
        {
            "Code": "{Contact Code}",
            "Name"; "{Contact Name}",
            "Email"; "{Email Address}"
        }
    ]
}]
```

```
[
{
    "Customer_ID": "CUSTA",
    "Name": "Acme Corporation",
    "Contacts": [{
        "Code": "CONTA",
        "Name": "Alison Acme",
        "Email": "aacme@acme.com"
    },{
        "Code": "CONTB",
        "Name": "Bob Acme",
        "Email": "aacme@acme.com"
    },{
        "Code": "CONTC",
        "Name": "Cindy",
        "Email": "aacme@acme.com"
    }]
},
{
    "Customer_ID": "CUSTB",
    "Name": "Other Corporation",
    "Contacts": [{
        "Code": "CONTF",
        "Name": "Roy",
        "Email": "aacme@acme.com"
    }, {
        "Code": "CONTT",
        "Name": "Karol",
        "Email": "aacme@acme.com"
    }]
}
]
```

6.2.1.1.2   Advanced JSON Example 2

If the JSON template does not have a top-level collection, there will be multiple calls to the Web API based on the results in the data table. There is one call per top-level entity. In the example below, the top-level entity is Customer. There will be two calls to the Web API, one for CUSTA and then one for CUSTB:

```
{
    "Customer_ID": "{Customer ID}",
    "Name": "{Customer Name}",
    "Contacts":   [
        {
            "Code": "{Contact Code}",
            "Name"; "{Contact Name}",
            "Email"; "{Email Address}"
        }
    ]
}
```

```
{
    "Customer_ID": "CUSTA",
    "Name": "Acme Corporation",
    "Contacts": [
        {
            "Code": "CONTA",
            "Name": "Alison Acme",
            "Email": "aacme@acme.com"
        },
        {
            "Code": "CONTB",
            "Name": "Bob Acme",
            "Email": "aacme@acme.com"
        },
        {
            "Code": "CONTC",
            "Name": "Cindy",
            "Email": "aacme@acme.com"
        }
    ]
}
```
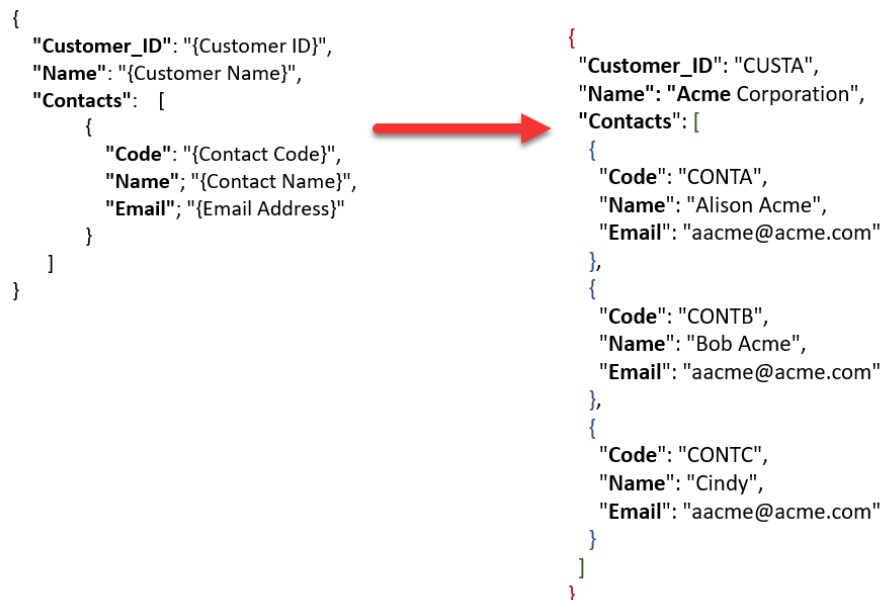
6.2.1.1.3   Advanced JSON Example 3

If the Web API action occurs in an array inside of JSON instead of a JSON template, users can combine advanced JSON and query strings to execute the parameter type POST for web APIs (for example: use *url?queryString={value that does not exist in JSON body}* as a parameter).

6.2.1.1.3.1   Example using Numeric {column#} to identify numeric value

o   *Numeric values in JSON to be replaced by using a double quotation: "key": "{value}#"*

```
1    {
2        "invoice": {
3            "id": "{InvId}",
4            "issueDate": "{InvIssueDate}",
5            "invoiceLine": [
6                {
7                    "id": "{InvLineId}",
8                    "invoicedQuantity": {InvLineQuantity},
9                    "invoicedQuantityUnitCode": "{InvLineQuantityUnitCode}",
10                   "lineExtensionAmount": {InvLineExtensionAmount},
11                   "taxTotal": [
12                       {
13                           "taxAmount": {InvLineTaxTotalTaxAmt},
14                           "taxSubtotal": [
15                               {
16                                   "taxableAmount": {InvLineTaxTotalSubTaxablAmt},
17                                   "taxAmount": {InvLineTaxTotalSubTaxAmt},
18                                   "taxCategory": {
19                                       "id": "{InvLineTaxTotalSubTaxCatID}",
20                                       "taxScheme": {
21                                           "id": "{InvLineTaxTotalSubTaxScmID}",
22                                           "idSchemeID": "{InvLineTaxTotalSubTaxScmIDID}",
23                                           "idSchemeAgencyID": "{InvLineTaxTotalSubTaxScmIDAgncyID}"
24                                       }
25                                   }
26                               }
27                           ]
28                       }
29                   ],
```

To manage numbers:

- Option 1 (<u>Selected</u>): Based on Query type, it will be integer or decimal
  {{ "lineExtensionAmount": "{InvLineExtensionAmount}#",}}
- Option 2: Separate type in order to transform in JSON using #
  {{ "lineExtensionAmount": "{InvLineExtensionAmount}#",}}
  {{ "lineExtensionAmount": "{InvLineExtensionAmount}#.##",}}
- Option 3: Separate type in order to transform in JSON using d (decimal) or n (numeric)
  {{ "lineExtensionAmount": "{InvLineExtensionAmount}n",}}
  {{ "lineExtensionAmount": "{InvLineExtensionAmount}d",}}

6.2.2   Web API Designer Authorization tab

On the Authorization tab, pick the authorization type. *If you are using an API key, do not use the Authorization Type setting and instead pass it using a key-value pair in the Parameters tab.*

- **No Auth**: Do not pass authorization credentials.
- **Basic Auth**: Enter the username and password you created for the API.
- **OAuth 1.0**: Please see the section below.
- **OAuth 2.0**: Please see the section below.

6.2.2.1.1   OAuth 1.0

OAuth 1.0 is an authorization method that provides access to third party resources to an HTTP service. OAuth 1.0 uses request and access tokens over HTTPS.

When selecting OAuth 1.0 as an authorization type, additional information is needed for authentication:

- **Callback URL**
- **Realm (Account ID)**
- **Consumer Key**
- **Consumer Secret**
- **Access Token (Token ID)**
- **Token Secret**
- **Verifier**
- **Signature Method**

6.2.2.1.2   OAuth 2.0

OAuth 2.0 is the industry-standard protocol for authorization which allows third-party applications to grant limited access to an HTTP service, either on behalf of a resource owner or by allowing the third-party application to obtain access on its own behalf.

OAuth2 requires the use of HTTPS for communication between the client and the authorization server because of sensitive data passing between the two.

When selecting OAuth 2.0 as an authorization type, additional information is needed for authentication:

- **Access Token**: Click the Request Token button at the bottom of the screen and a token is sent from the third-party application and it is automatically populated. Access Token is masked. The token is saved by selecting **the Use Saved Access Token for All Requests** check box and used for all requests to this API during event execution. If unchecked, a new Access Token will be generated for each new request.
- **Grant Type:** The method of acquiring an access token. Choose one of the following types:
  - **Authorization Code:** Use when the client is a web server. Allows you to obtain a long-term access token since it can be renewed with a refresh token if the authorization sever enables it. The following information is required for this grant:
    - Callback URL (This is generated by KnowledgeSync.)
    - Auth URL
    - Access Token URL
    - Client ID (masked)
    - Client Secret (masked)
    - Scope
    - State
  - **Authorization Code (with PKCE):** Proof Key for Code Exchange, use to authenticate native or mobile application users. Because a client_secret isn't required, this reduces security risks. The following information is required for this grant:
    - Callback URL
    - Auth URL
    - Access Token URL
    - Code Challenge Method
    - Code Verifier
    - Client ID (masked)
    - Client Secret (masked)
    - Scope
    - State

- o **Implicit:** Use when the client is running in a browser and using a scripting language, e.g., JavaScript. This grant type does not allow for the issuance of a refresh token. The following information is required for this grant:
  - Callback URL
  - Auth URL
  - Access Token URL
  - Client ID (masked)
  - Client Secret (masked)
  - Scope
  - State
- o **Password Credentials:** Use when the client has been developed by the same authority as the authorization server. With this type of authorization, the credentials (and thus the password) are sent to the client and then to the authorization server. The following information is required for this grant:
  - Access Token URL
  - Username
  - Password
  - Client ID (masked)
  - Client Secret (masked)
  - Scope
- o **Client Credentials:** Use when the client is himself the resource owner. There is no authorization to obtain from the end-user. The following information is required for this grant:
  - Access Token URL
  - Client ID (masked)
  - Client Secret (masked)
  - Scope
- **Callback URL**
- **Client Authentication**: Chose one of the following types for sending client authentication:
  - o Send as Basic Auth header
  - o Send client credentials in body
  - o Send Query String

Check the box for **Use Saved Access Token for All Requests** when creating a new connection with credentials that will be used for several events and eventpaks. When creating an event, users can select from a list of established connections to auto-populate the credential information.

6.2.3   Web API Designer Parameters tab
On the Parameters tab:

- **Variables**: Search for a system variable here, and then double-click it to insert the variable into the Value field.
- **Parameter Type**: Choose what works best for you:
  - o **Get or Post**: This behaves differently based on the verb. In both cases, name and value are URL-encoded.
    - **Get**: Parameters are appended to the URL in the form URL?name1=value1&name2=value2.
    - **Post or Put**: Parameters are sent in the request body.
      - Users can combine advanced JSON and query strings to execute POST for Web APIs, e.g. *url?queryString={value that does not exist in JSON body}*
  - o **Query String**: This works like "Get or Post", except it appends the parameters to the URL in the form URL?name1=value1&name2=value2 regardless of the selected verb.
  - o **URL Segment**: When the request executes, this parameter type replaces a {bracketed} variable in the URI field having {Name} with the Value.

- o **Http Header**: Adds the parameter as an HTTP header sent along with the request. The header name is the name of the parameter and the header value is the value.
- o **Request Body**: If this parameter is set, its value is sent as the body of the request. *Only the first Request Body parameter is accepted.*

> o *Request Body does not work on Get requests, as they do not actually send a body. If you have "Get or Post" parameters as well, they will <u>replace</u> the Request Body parameter.*

- o **Cookie**: Adds the parameter to the list of cookies sent along with the request.
- o **Request Root**: If this parameter is set, its value is sent as the root of the request.
- **Key**: The key portion of the key-value pair.
- **Value**: The value portion of the key-value pair.

You can also insert system variables in {brackets} that were defined in previously executed web API triggers. This example shows how you can include a "summary_shorts" variable as a value. Your custom variables will not display in the Variables list, but you can manually type them.

- **Event Parameter**: Specifies where the web API definition will get the value for this parameter:
  - ▪ Checked: KnowledgeSync will get the value from the event that uses this Web API as a trigger. To set the value, go to the Trigger properties section in the event's screen. If the event does not provide a value, KnowledgeSync will use the value entered in this web API definition as the default.
  - ▪ Unchecked: KnowledgeSync will get the value from the Value field within the Web API definition.
- **Combine All parameters into a JSON (XML) Request Body**: Check this option if your web API requires all parameters to be combined into a single JSON or XML object in the body of the request rather than sent as individual parameters. (The parameter shown defaults to the Format selected on the Definition tab.)

> *Note: Web API filters use SQLite syntax on all calculated fields on the Parameters and Response tabs. Users can enter their own filters using <u>SQLite syntax</u>.*

6.3   Web API Raw Response Tab

Raw JSON or XML responses from the API is displayed on the Raw Response tab. This tab contains a simplified view of the response structure. Users can directly add values from the Response tab and KnowledgeSync formats the path appropriately by detecting the data type of value (simple, list/array, object, file) from the response.

6.4   Web API Designer Response Tab

The Response tab calls the REST API using your settings from the Definition tab. A tree structure representing the JSON is displayed. The path is automatically filled in and represents the branch or node of the JSON tree that is selected.

- **Response of API call**: The response is displayed here in native JSON format. Click on headers to review the header information received from the API call:
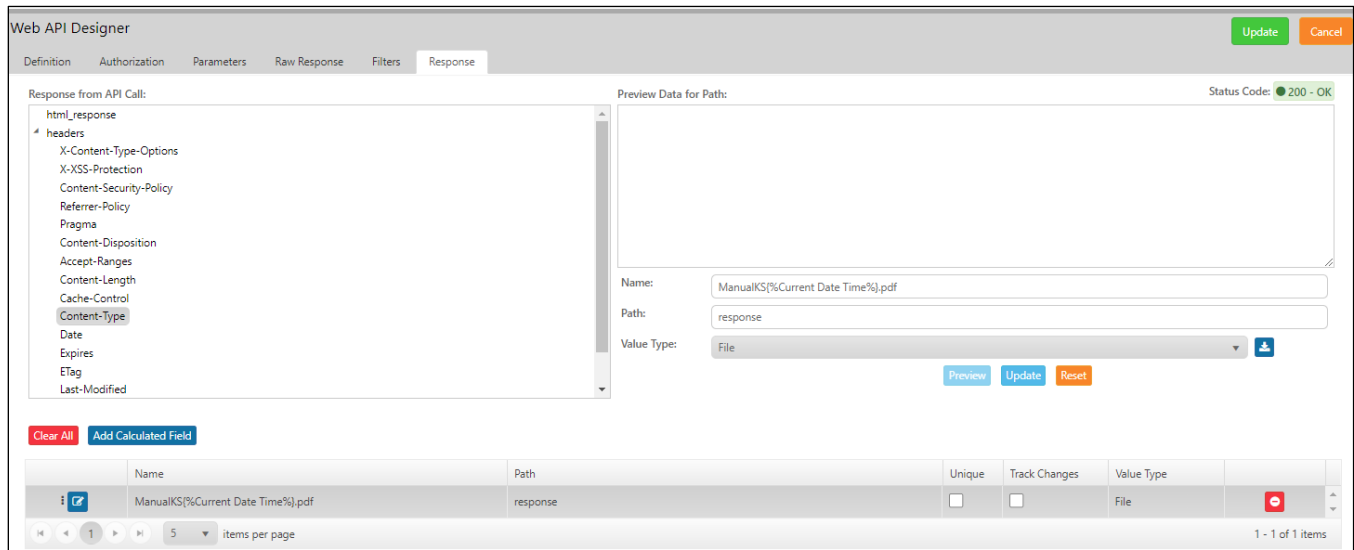
- **Preview of Data for Path**: The path you chose is displayed for confirmation, and then the value of the path is displayed. Select the data to preview in the Response from API Call module and then click the **Preview** button.
- **Name**: The name you want for the system variable. When working with a downloaded file, the name of the file from the response is automatically inserted. When working with the file later, the name can be dynamic or use Calculated Fields to hardcode a name to use with every event.
- **Path**: When KS ITC receives the REST API response, it will generate a list of possible JSON paths. Pick from the available path or type your own.

    A common path in KS is to replace "[0]" with an asterisk "[*]" to cause KS ITC to retrieve a list, such as changing "results[0].display_title" to "results[*].display_title".

- **Value Type**: Assign the data type that KS ITC should use for this variable: number, string or date.

6.5   Filtering Results
You have two options for filtering results from a REST API:

If the REST API supports querying, refer to the documentation for creating such a query. Examples may include:
  - o   https://restdb.io/docs/querying-with-the-api
  - o   https://graphql.org/

If the REST API does not support querying, you can pass the dataset returned from the REST API trigger to a second database trigger and use the SQL's where clause to filter the results.

The examples below show both methods. Both examples are high level since they skip individual clicks and data entry and show the material changes needed to accomplish the task. See the relevant sections of this manual for more detailed instructions.

6.5.1   Filtering API Example: Filtering New York Times Movie Reviews
The New York Times Movie API allows you to pass filter using several methods. This permits you to have the API perform the filtering. No further filtering is required once the data is in KnowledgeSync.

6.5.1.1   Movie Title
The "query" parameter accepts the value of the movie title, as shown below:
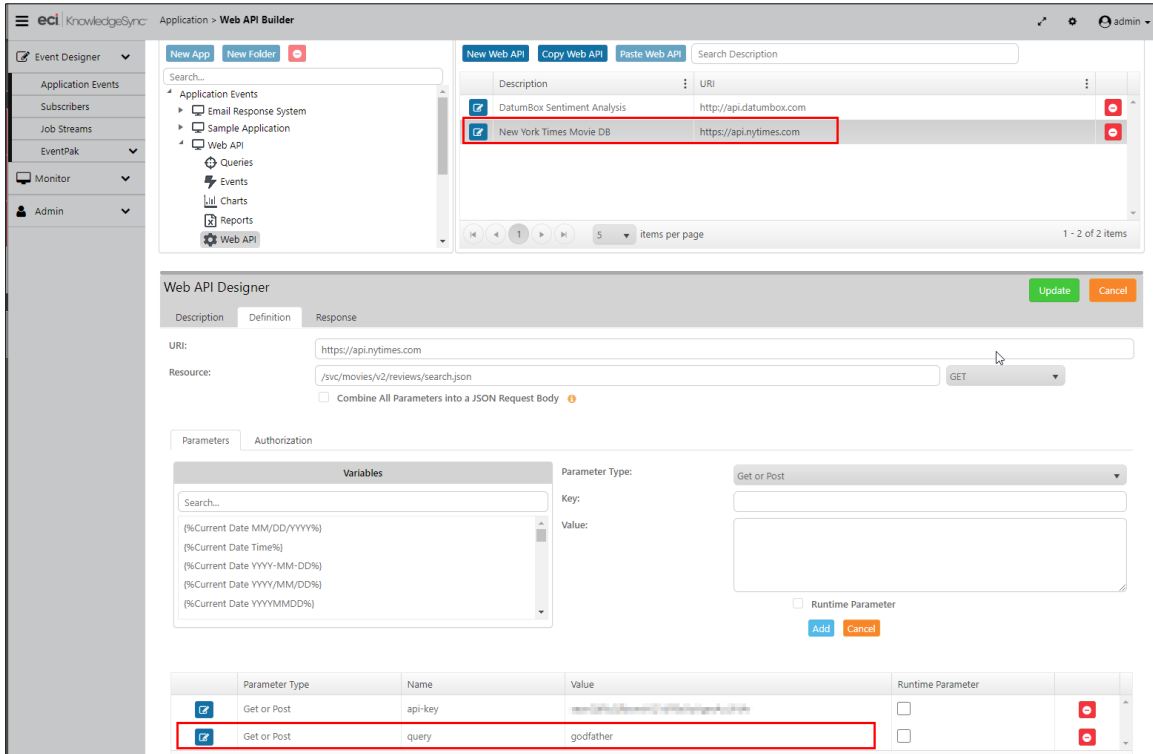
You could substitute a KnowledgeSync variable in the place of "godfather".

6.5.1.2   Movie Title and Opening Date
Additional filtering can be performed on many different fields on this API. This example finds movies with "big" in the title opening during the 1980s.
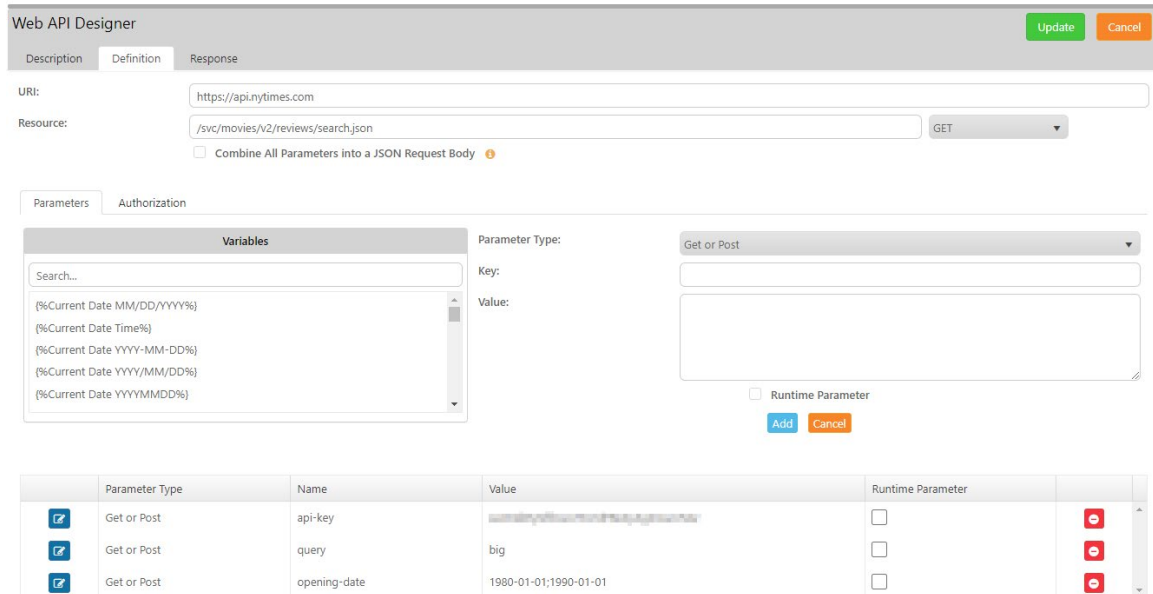
6.5.2   Non-Filtering API Example: Filtering on Date for Lasso CRM
In this example, we'll get a list of properties from Lasso CRM. We want to email property owners three days before their property possession date. In this case the REST API does not support filtering, so to do this we'll get a list of properties from the API, use the database engine to filter the list, and then send an email.

6.5.2.1   Getting Keys in Three Days

6.5.2.1.1   Create REST API Trigger

Create a new Web API object. From the example below, you see the API response includes dates.
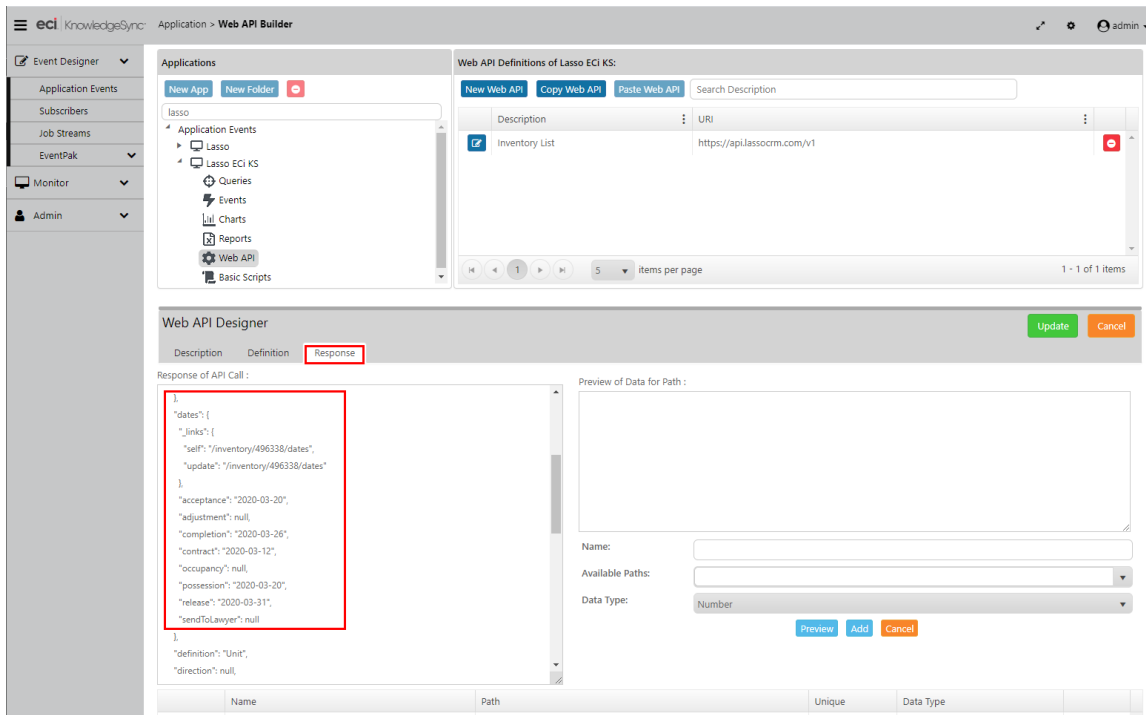
6.5.2.1.2   Create Database Connection

Create a database connection to use in the next step. We connected to KS_Object, but the actual database doesn't matter since we'll only use the query engine for our date comparison, and we won't read or modify any data tables.
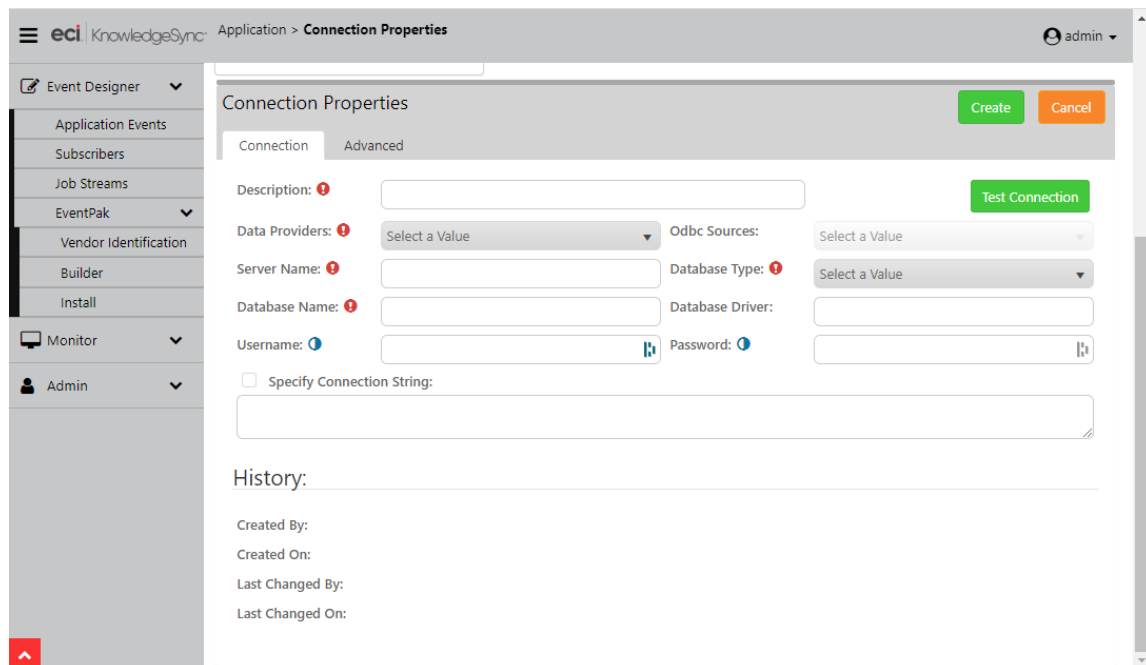
6.5.2.1.3   Create Query

Create a query using any database table. In the example below, we used the "admin" table.

- Use the database connection from the previous step.
- Add a calculated column with value = 1. This column isn't used, but KnowledgeSync requires at least one field in a query.
- Set Return Distinct Rows of Data Only = Checked.
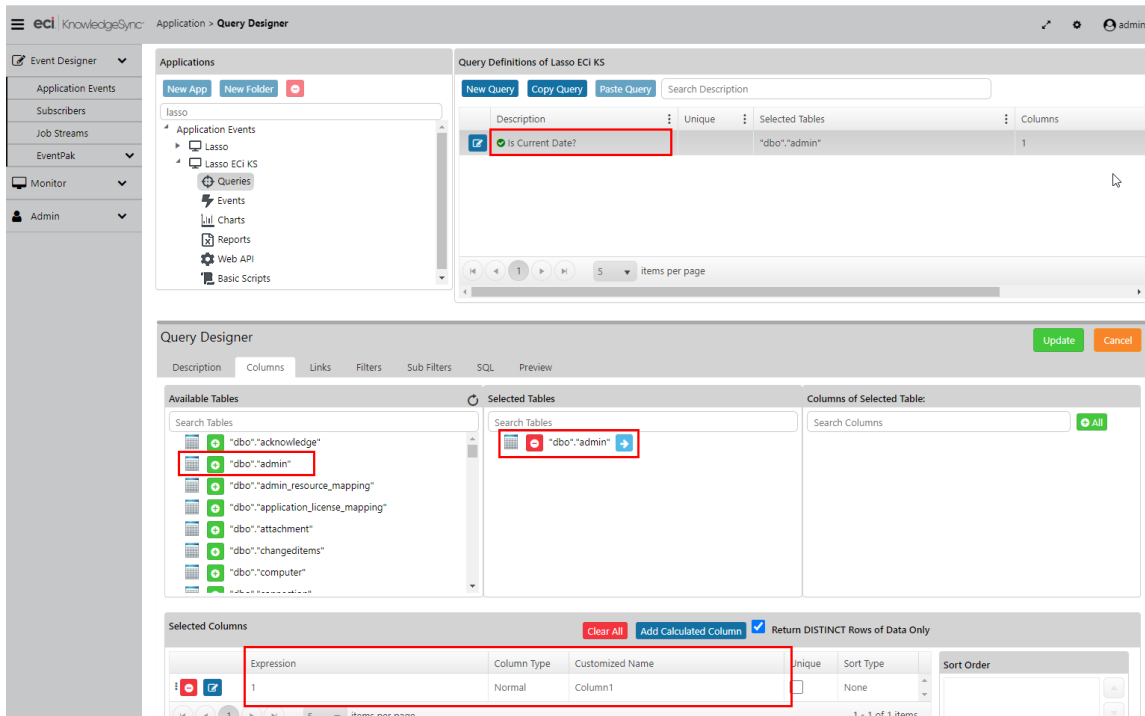
- Go to the Filters tab and create a SQL filter statement. This statement is:

```
DATEADD(DAY,-3,'{%Current Date%}')
```

This statement is SQL language intended for SQL Server. It subtracts three days from the current date. It will be used in the event in next step to compare with the property possession date.

There are several settings in addition to the SQL statement, such as Type = Date.
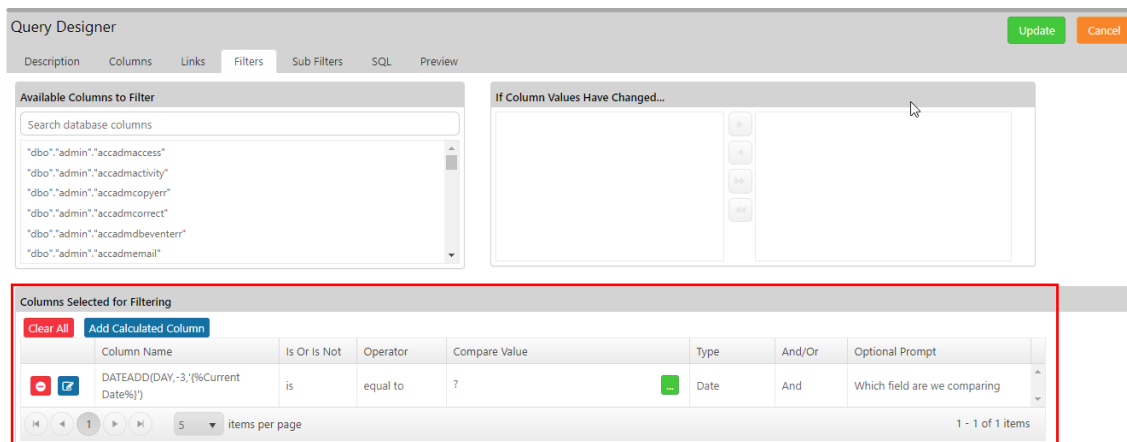
6.5.2.1.4   Create Event
Create an event.

- Add two triggers, which must appear in this order:

1. Inventory List (Gets the list of properties)
2. Is Current Date? (Compare each item in the Inventory List using their possession date with the date that is "three days from now")
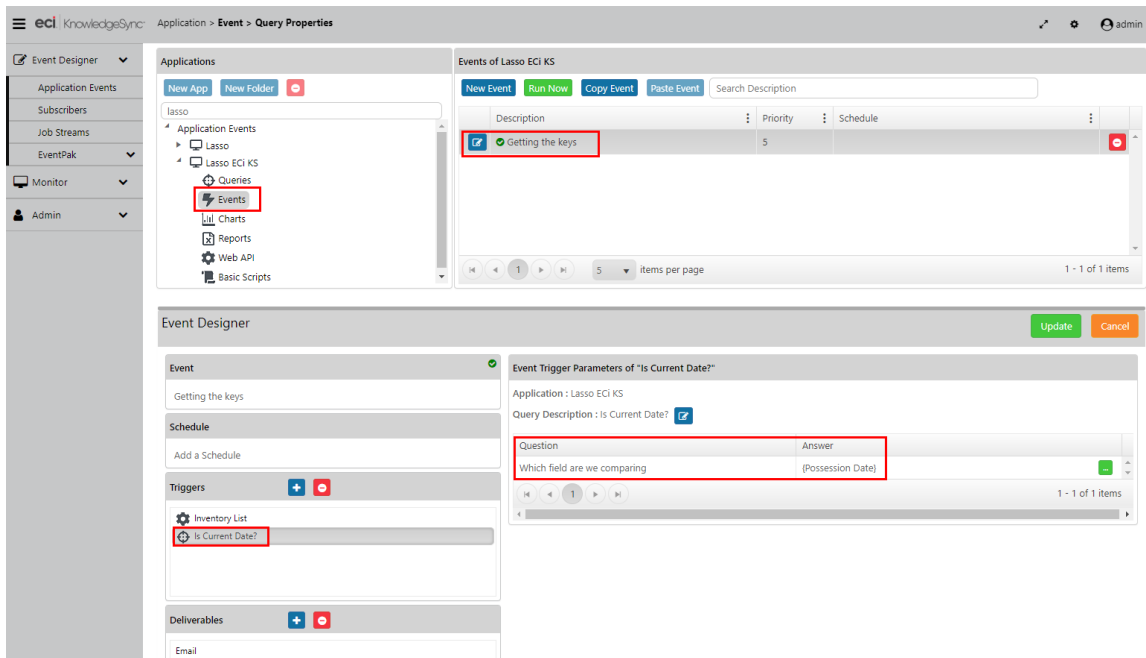
- For the second trigger, set the Parameter Value to "{Possession Date}".

Variables from the REST API trigger will not appear in the "Available Values for Use:" box, but if you type the value, it will be accepted and work.

6.5.2.1.5   Create Email

Create a standard KnowledgeSync email action. We used a BEGIN/END REPEAT to group properties.
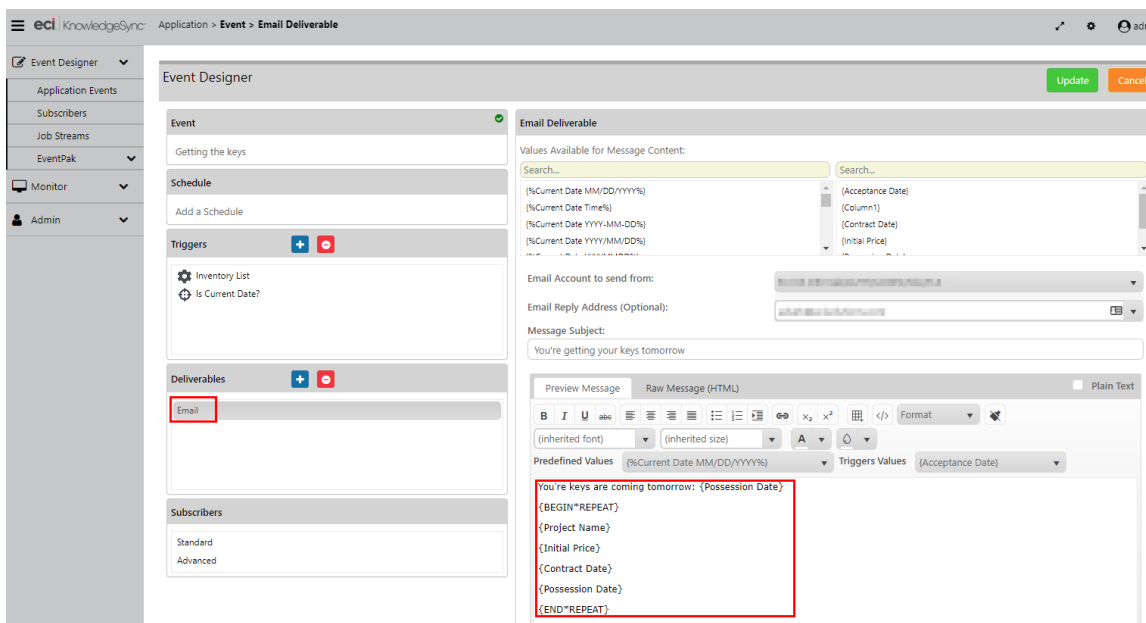
6.5.2.2   Contract Date within 7 Days of Possession Date

6.5.2.2.1   Create REST API Trigger

Create a new Web API object like the same step in the previous example.

For reference, we will compare the two values from the response shown:
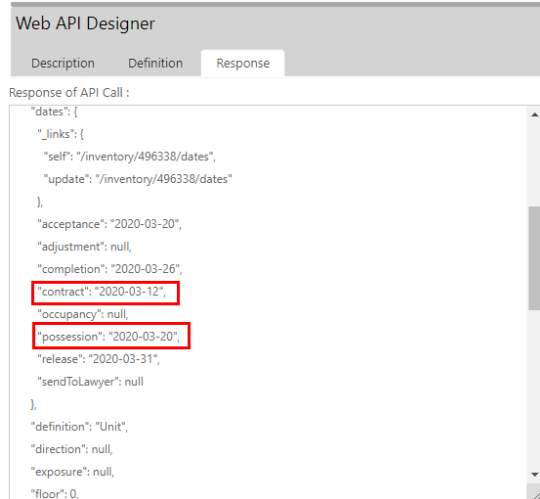
6.5.2.2.2   Create Database Connection

Create a database connection like the same step in the previous example.

6.5.2.2.3   Create Query

Create a query using any database table. In the example below, we used the "admin" table.

- Use the database connection from the previous step.
- Add a calculated column with value = 1. This column isn't used, but KnowledgeSync requires at least one field in a query.
- Set Return Distinct Rows of Data Only = Checked.
- See §3.1.3 for a sample screen shot for the progress to this point.
- Go to the Filters tab and create a SQL filter statement. This statement is:

    7

    This statement is a single numeric value that will be later combined with SQL language intended for SQL Server.
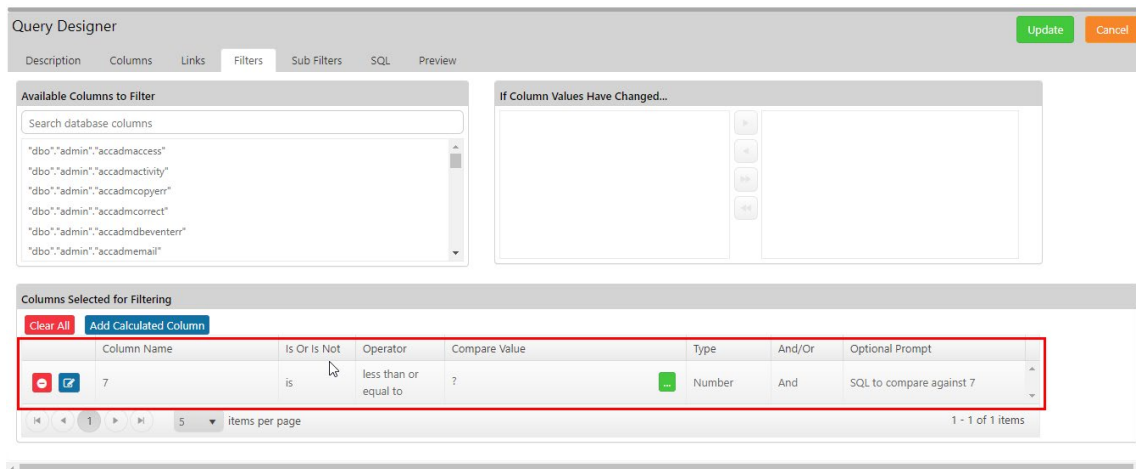
6.5.2.2.4   Create Event
Create an event.

- Add two triggers, which must appear in this order:

1. Inventory List (Gets the list of properties)
2. Date 1 Within 7 Days of Date 2? (Compare each item in the Inventory List using their possession and contract dates with the value 7 in "Date 1 Within 7 Days of Date 2")

6.5.2.2.5   Create Email
Create a standard KnowledgeSync email action like the same step in the previous example.
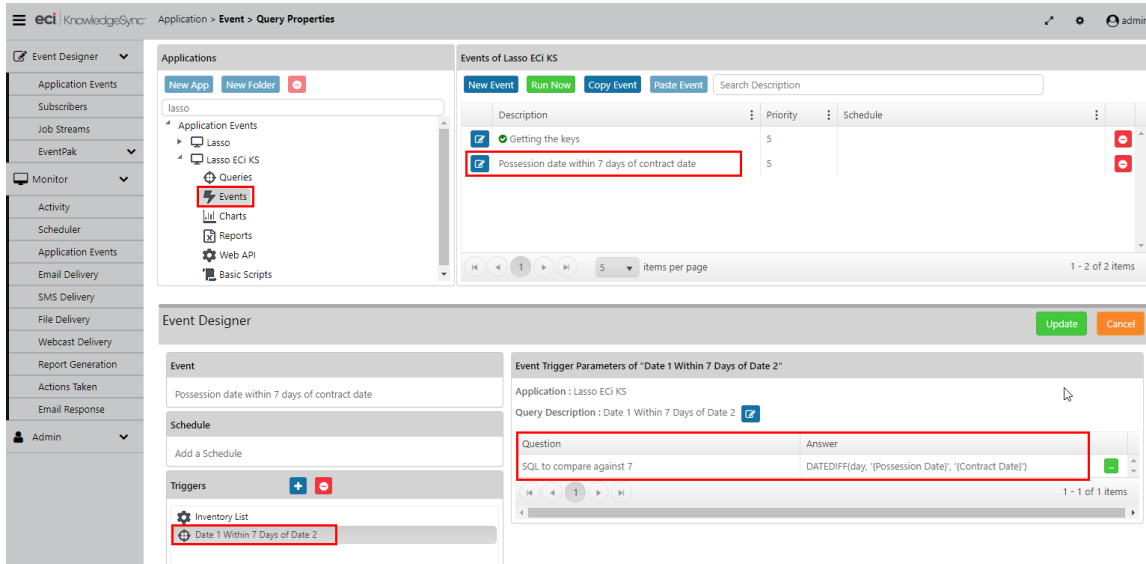
- For the second trigger, set the Parameter Value to "DATEDIFF(day, '{Possession Date}', '{Contract Date}')".

We are defining the "where" portion of the SQL statement here in the event, which is then compared against the value 7 when KnowledgeSync combines the values, as in SELECT … WHERE 7 = DATEDIFF(day, '{Possession Date}', '{Contract Date}').

Variables from the REST API trigger will not appear in the "Available Values for Use:" box, but if you type the value, it will be accepted and work.

7 Visual Basic (VB) Scripting

Among KnowledgeSync's capabilities is its ability to use Visual Basic (VB) scripts in two ways:

- As a <u>trigger</u> to an event (i.e., to determine if certain conditions exist). For example, to see if a specific file exists in a folder
- As a <u>deliverable</u> to an event (typically used to add or update information within application databases). For example, to change an overdue customer's credit status from 'Good' to 'On Hold'

This component should be used only by those individuals who are familiar with VB syntax and have expertise in writing VB scripts.

> *KnowledgeSync Support will answer questions about how the VB Scripting works and will also provide VB script samples. However, KnowledgeSync Support does not provide custom script design support or debugging services. Please contact your Business Partner for details on these services.*

7.1 Sample Scripts

Sample scripts are available for your review at the [KnowledgeSync Software Install Upgrade Reference Library](link).

7.2 Scripting Tools: Version 9 (& Prior); Version 10

In version 9 (and previous versions of KnowledgeSync), the embedded VB Scripting Tool was published by Cypress Enable. As of version 10, the VB scripting tool is the WinWrap VB Scripting engine (http://winwrap.com/web2/). Cypress Enable scripts were designed and debugged within KnowledgeSync; WinWrap scripts are designed and debugged outside of KnowledgeSync, and only the name and location of the script file is stored in KnowledgeSync.

All version 9 KnowledgeSync sites that are using Cypress Enable scripts and choose to upgrade to version 10 code will have their existing scripts auto-converted to the new syntax required by WinWrap Basic. If you have issues running any Cypress Enable scripts under v10, please [contact the KnowledgeSync Support](link) department.

With KnowledgeSync version 10.02 and newer, we support VB.NET scripts in additional to VB. The WWB.NET Scripting language enables WinWrap script writers to use VB.NET style syntax in their scripts. WWB.NET Scripting Language scripts can access and utilize Microsoft .NET Framework's vast array of public classes and interfaces.

7.3 COM and .NET

WinWrap allows two styles of scripting – COM and .NET. The difference is indicated in code files by the header and style of scripting specified.

- `'#Language "WWB.COM"` compatible with VB6, VB scripting style syntax (this is the current implementation to support scripts from KnowledgeSync V9)
- `'#Language "WWB.NET"` compatible with VB.NET style syntax.

KnowledgeSync V10 identifies the script type by the file extension: *.vb* indicates that the file uses `'#Language "WWB.NET"` and *.bas* or *.bat* indicates `'#Language "WWB.COM"`.

WinWrap Basic documentation can be found at http://www.winwrap.com/web2/basic/#!/ref/doc-references.

7.4 Create a Script

Before you create a script, determine whether the script will be used as a trigger or as a deliverable.

If to be used as a trigger, be sure the script retrieves all the data fields your event requires. Only fields that are retrieved by the script trigger will be able to be used in an event's alert messages, passed into the event's reports, and used in any of the event's "Actions" deliverables.

If the script is to be used as an event's "Action" deliverable, be sure that the event's trigger(s), retrieve all the data that is needed by the script deliverable.

To define a script within KnowledgeSync, expand the "All Events" branch, expand the application you're working with, click on the "Basic Scripts" sub-branch and click on the "New Script" button.
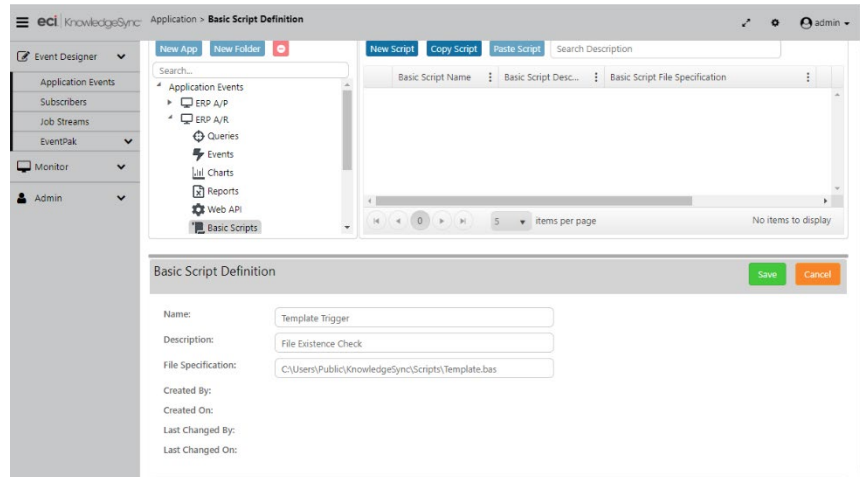
WINWRAP VB SCRIPT DEFINITION

Name. Specify a brief descriptive name for the script. It is suggested that you also include in this name whether the script is to be used as a "trigger" or as a "deliverable".

Description. The name you specify is auto-copied into the script description field, in which you may enter a more detailed purpose of the script.

File Specification. Specify the name and location of your script; you may wish to store them in the KnowledgeSync "Scripts" folder. Be sure not to reference mapped drives in this field.

Save and close the script definition window.

7.5   Sample Script as Deliverable
A script selected as an event's deliverables will look like the following:
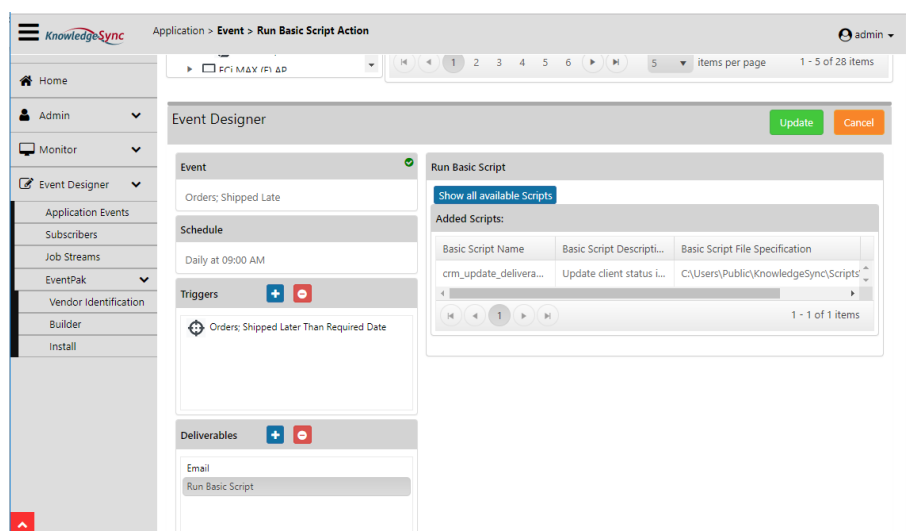
LINKING A DELIVERABLE SCRIPT TO AN EVENT

7.5.1 Deliverable Script Part I: Name & Purpose

A "deliverable" script should begin with the identification of the script's name, purpose, and author:

```
' Script Name          : vs_sample_action.bas
' Script Author        : Development
' Script Purpose       : sample showing script used as action
' Script Creation Date : 9/14/2019
' Script Modified by   :
' Script Modified Date :
' Script Modifications :
```

7.5.2 Deliverable Script Part II: Retrieve "Connection" Data

The second part of a "deliverable" script retrieves information about the KnowledgeSync application the event belongs to and its (database) connection:

```
Declare Function KSGetEvent App ( ByVal s As String ) As String
`
' This function retrieves specific data about the event that
' has triggered.
'
' The available values that can be passed to KSGetEvent in order
' to retrieve data are as follows:
'
' lookup             Returns the Lookup key as defined in
'                    the Event Manager
' application The application that the event belongs to
' description The description of the event
' id                 The id of the event
' dns                The ODBC datasource name of the application
' username           The username for the connection of the application
' password           The password used for the connection
```

7.5.3 Deliverable Script Part III: Retrieve "Connection" & Alert Message Data

The third part of a "deliverable" script retrieves information about the triggered event's alert message:

```
Declare Function KSGetPackage App ( ByVal s As String ) As String
'
' This function retrieves specific data about the package that was
' created for delivery. The available values that
' can be passed to KSGetEvent in order to
' retrieve data are as follows:
'
' email subject       The subject of the email created for delivery
' email message       The email message text created for delivery
' text message        The text message text created for delivery
' webcast subject   The webcast subject text created for delivery
' webcast message   The webcast message text created for delivery
```

7.5.4 Deliverable Script Part IV: Retrieve SQL Syntax

The fourth part of a "deliverable" script retrieves the triggered event's SQL syntax:

```
Declare Function KSGetSQL App ( ByVal l As Long ) As String
'
' This function retrieves the SQL from the query that was used to
' trigger the event.
' You pass a number of the query used.
' For example if the event has only one (1)
' query, then you would pass the number 1
' Example: mysql=ksgetsql(1)
' retrieves the sql from the first query
```

7.5.5   Deliverable Script Part V: Retrieve Query (Trigger) Columns

The fifth part of a "deliverable" script retrieves data from the event's queries/triggers:

```
Declare Function KSGetData App ( ByVal s As String ) As String
'
' This function retrieves columns of data from the queries that
' were used to trigger the event.
' You pass the customized column name as defined in the query builder to
' retrieve that information.
' Example: print ksgetdata("companyname") ' prints
' company name from the query data
```

7.5.6   Deliverable Script Part VI: Retrieve Script Completion Status

The sixth part of a "deliverable" script retrieves the completion status code and status description of the executed script:

```
Global KSReturnStatus As Long
'
' Set this value if your script if you want to return an error status number.
' If you don't set this value, then the script is assumed to have
' worked successfully and is stamped completed by the system.
'
Global KSReturnMessage As String
'
' Set this value in your script of you want to return an error message
```

7.5.7   Deliverable Script: Sample Output

You can use syntax such as the following to show the results of the executed script:

```
Open "c:\vs_sample_action.txt" For Append As #1

' Print out some event specific information

Print #1, "ID:" & ksgetevent("id")
Print #1, "Desc:" & ksgetevent("description")
Print #1, "App:" & ksgetevent("application")
Print #1, "Look:" & ksgetevent("lookup")
Print #1, "DNS:" & ksgetevent("dns")
Print #1, "username:" & ksgetevent("username")
Print #1, "password:" & ksgetevent("password")
Print #1, "Notes:" & ksgetevent("notes")

' Now print out some content delivery package info

Print #1, "Here is some the package info:"
Print #1, "Subject:" & ksgetpackage("email subject")
Print #1, "Message:" & ksgetpackage("email message")
Print #1, "Some other info:"

' Here is the SQL that was used to generate the event

Print #1, "SQL:" & ksgetsql(1)

' Now print out some of the query info

Print #1, "Company:" & ksgetdata("companyname")
Print #1, "Contact:" & ksgetdata("contactname")
Print #1, "Ordered:" & ksgetdata("orderdate")
Print #1, "Orderid:" & ksgetdata("orderid")
Print #1, "Total$:" & ksgetdata("total")
Close #1
```

7.6   Sample Script as Trigger

A "trigger" script contains instructions to check for a specific condition of data and then identify those elements of data that need to be passed into an event. (A single script <u>cannot</u> be used as both a trigger and a deliverable.)
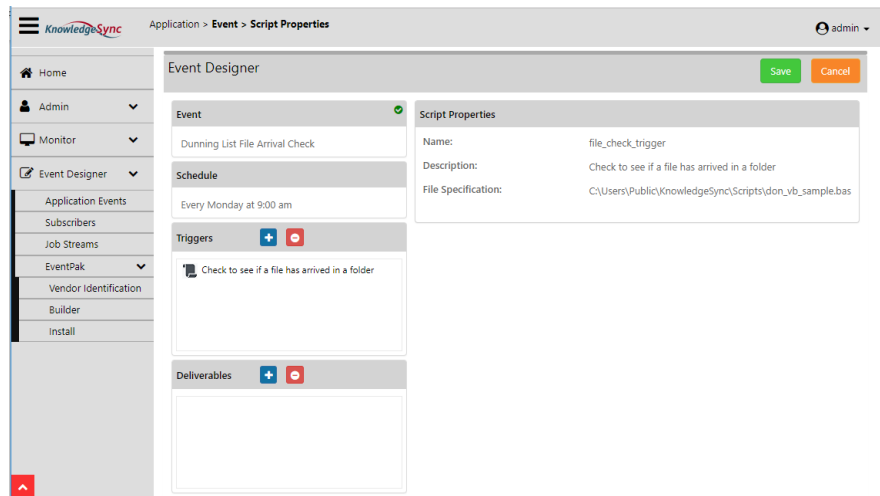
<p align="center"><small>LINKING A TRIGGER SCRIPT TO AN EVENT</small></p>

A trigger script can perform whatever sort of checking you need it to – monitoring the status of hardware devices, checking the revision dates of files, scanning for critical processes that must be running, or retrieving data from a database.

7.6.1   Trigger Script Part I: Name & Purpose

Like a "deliverable" script, a "trigger" script should begin with the identification of the script's name, purpose, and author:

```
' Script Name         : vs_sample_diskspace.bas
' Script Author       : Development
' Script Purpose      : Check available diskspace and trigger event
' Script Creation Date : 2/20/2001
' Script Modified by   :
' Script Modified Date :
' Script Modifications :
```

7.6.2   Trigger Script Part II: Trigger Function

The second part of a "trigger" script enables you to call the event for which the script is acting as a trigger:

```
Declare Function KSTrigger( ByVal ukey As String) As Boolean
'
' This function triggers the event in the Application Server. This function
' can be called when the script is being used instead of a query
' and you want to trigger an event.
'
' The parameters are as follows:
'
' ukey  The data that uniquely identifies this triggered item.
'       Passed as a string. You may pass an empty string if
'       you do not want KnowedgeSync to keep track of triggered items.
```

7.6.3   Trigger Script Part III: The Trigger Script

The third part of a "trigger" script is the script itself; in the following example, a script is used to retrieve the amount of free space on a disk.

```
Declare Function GetDiskFreeSpace Lib "kernel32" Alias "GetDiskFreeSpaceA" _
(ByVal lpRootPathName As String, lpSectorsPerCluster As Long, lpBytesPerSector As Long,
lpNumberOfFreeClusters As Long, lpTtoalNumberOfClusters As Long) As Long

Dim strDrive As String
Dim lngSectorsPerCluster As Long
Dim lngBytesPerSector As Long
Dim lngNumberOfFreeClusters As Long
Dim lngTotalClusters As Long
Dim lngSts As Long
Dim DS
Dim IntLoop As Integer
Dim dblSpace As Double

'These variables can be seen from the event email message screen because they are global
'By declaring these, they can be used as content when we send email, faxes or pages...

Global DiskDrive As String
Global FreeSpaceBytes As Double
Global FreeSpaceMegs As Double
Global FreeSpaceFormatted As String
Global FreeSpaceMessage As String
Global MinMegs As long

MinMegs=50000          'This is our threshold. Less than 500 MB is a problem for us.

For IntLoop = 67 To 90
        strDrive = Chr$(IntLoop) & ":\"      ' Ensure path is at the root.
        lngSectorsPerCluster=0
        lngBytesPerSector=0
        lngNumberOfFreeClusters=0
        lngTotalClusters=0
        lngSts = GetDiskFreeSpace(strDrive, lngSectorsPerCluster, lngBytesPerSector,
lngNumberOfFreeClusters, lngTotalClusters)
        If lngSts <> 0 Then
        FreeSpaceBytes = CDbl(lngSectorsPerCluster) * CDbl(lngBytesPerSector) *
CDbl(lngNumberOfFreeClusters)
        FreeSpaceFormatted = Format$(FreeSpaceBytes, "###,###0")
                FreeSpaceFormatted = FreeSpaceFormatted & " bytes are free."
                FreeSpaceMegs = FreeSpaceBytes / 1000000
                If FreeSpaceMegs < MinMegs Then   'Less than MinMegs megs is BAD.
                        DiskDrive=chr$(intloop) & ":"
                        If DiskDrive <> "D:" Then  'Don't check the cdrom drive. * change this to
your CDROM drive.
                                FreeSpaceMessage = "Disk " & DiskDrive & " has less than " &
MinMegs & "mb free."
                                x= KSTrigger(DiskDrive)     'We have low space on this drive
                        End If
                End If
        End If
Next IntLoop

KSReturnMessage=0          'Clean as a whistle.

End     'End of Script'
```

8   Monitoring Events

The KnowledgeSync Monitor lets you keep track of:

- The status of the application's servers
- Events that are scheduled to be checked and have been triggered
- Alerts that are queued to be delivered, and have been delivered
- The text of the outgoing email alert messages
- The status of forms, documents, and reports queued for delivery
- The status of workflow actions queued for execution
- The status of any system errors

The Monitor can also be used to "correct" or delete the information that appears within it, based on a KnowledgeSync user's authorizations and permissions.

8.1   The Watchdog EventPak

KnowledgeSync can monitor itself, and there is a collection of events that perform this function. It is referred to as the "WatchDog EventPak" and it may be downloaded from the KnowledgeSync downloads page. The Watchdog monitors and reports on:

- Event errors
- Alert delivery errors (e.g., an invalid phone number)
- Too many alerts
- Too many alerts to one person
- Form, document, or report generation errors
- Workflow errors
- Alerts or actions pending for more than 'x' minutes

8.2   The Monitor

When you click on the Monitor from the main KnowledgeSync window, the first branch ("Activity") shows you the status of KnowledgeSync's servers:

KNOWLEDGESYNC MONITOR : "ACTIVITY"

8.2.1   Activity Folder

This window is a listing of all of KnowledgeSync's servers, whether they are enabled to run (the first column, which can be toggled on or off), the "status" column (showing whether a server is currently running), and the "details" column (showing the text of any error messages encountered by the server).

V10.6.5 User Manual

You can optimize system resources by disabling (toggling "off") one or more servers if your configuration of KnowledgeSync will not be using that server's functionality. (For example, if you will not be sending alerts via web dashboard (webcasting), you can toggle this server off so that it uses no system resources.

8.2.1.1 Server Status
This displays the current status of the application's servers. Possible statuses are:

- **Shutdown**: The server is not running at this time. This could be due to the application service not running, the individual server being disabled, or an error that caused the shutdown of the server.
- **Idle**: The server is running but is not processing any data.
- **Processing**: The server is processing data.
- **Startup**: The server is starting up. Startup mode should not last for more than a few seconds; if it does, try re-starting the application's Windows service.
- **Linking**: Subscriber linking (see "Subscribers" chapter) is in progress.

KnowledgeSync uses the following servers:

- **KnowledgeSync**: This server initiates all of the other servers.
- **Scheduler**: This server schedules events, alerts, reports, and workflow.
- **Application Event**: This server executes events.
- **Email**: This server delivers email messages.
- **SMS**: This server delivers text messages.
- **Copy**: This server copies and delivers files and reports via FTP/SFTP.
- **Report**: This server generates Crystal and SSRS reports, forms, and documents.
- **Webcaster**: This server delivers dashboard (webcast) messages.
- **Action**: This server executes workflow actions.
- **Email Response**: This server handles the processing of incoming email monitored by KnowledgeSync

If any of the first three servers are shutdown or paused, KnowledgeSync will not be able to check for events, send alerts, generate reports, or execute workflow.

If any of the notification servers (i.e., email, copy, text or webcast) are shutdown or paused, KnowledgeSync will not be able to send alerts via those methods.

If the "Action" server is shutdown, KnowledgeSync will not be able to execute workflow actions and/or trigger updates into one or more application databases.

8.2.1.2 Current User Logins
This option tells you who is logged into KnowledgeSync, including each person's login name, login date/time, as well as their platform and browser versions.

8.2.2 Scheduler
The "Scheduler" folder shows you what events are due to be submitted for execution next.
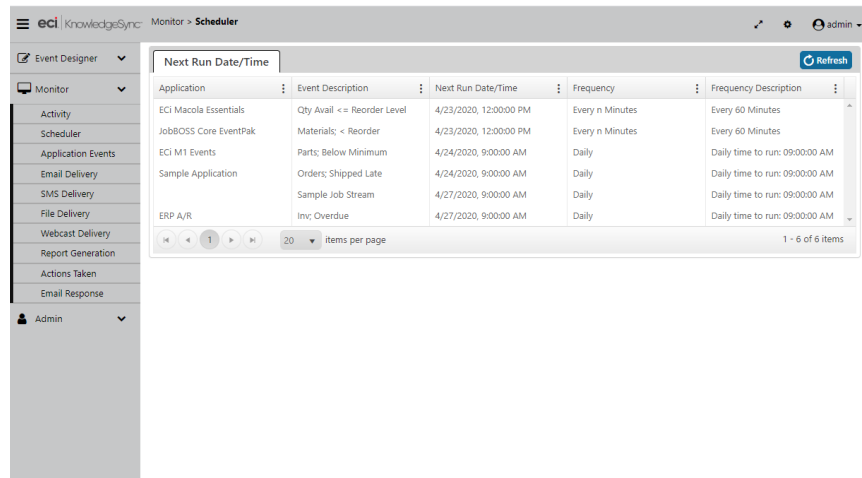
NEXT SCHEDULED RUN DATE/TIMES

For each event you will see its application name, event description, next scheduled run date/time, and the event's schedule frequency.

8.2.3   Application Events

The "Application Events" folder lets you to track the status of events that are due to run, have already run, or have failed to run successfully because of an error.
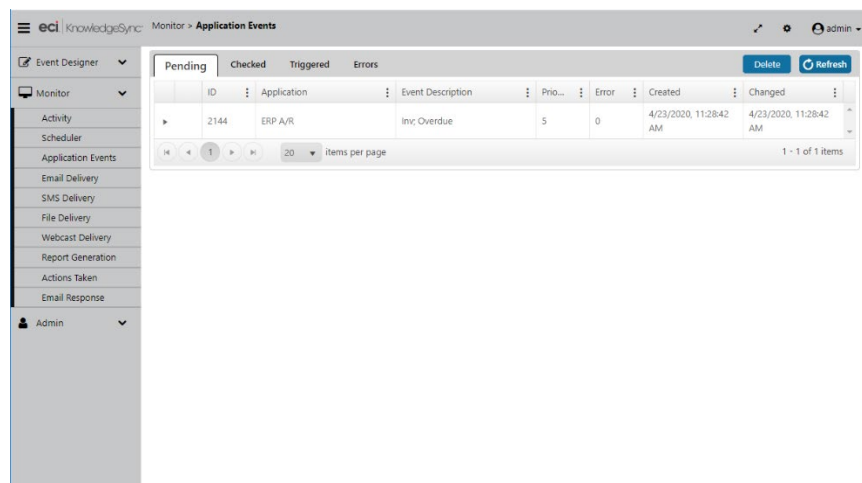
APPLICATION EVENTS: PENDING, CHECKED, TRIGGERED & ERRORS

Application events are divided into four categories:

- **Pending**. An event is pending because it either is about to be submitted or has been submitted and is in an error state.
- **Checked**. A checked event is one that was successfully executed. It may or may not be "triggered", depending on whether its conditions were met.
- **Triggered**. A triggered event is one that was successfully executed and whose conditions were met. A single run of an event may result in multiple triggered records appearing in the Monitor.
- **Errors**. An event shows up under "errors" if it was unable to be successfully checked. Such an event automatically is re-queued under "Pending".

8.2.3.1 Triggered Records

When an event successfully runs, it will appear once in the "Checked" branch and may appear multiple times in the "Triggered" branch. Consider the following event:

Send an alert to each support rep about any high priority call that has been open for longer than four hours.

There may be multiple call records that meet this event's criteria and the event may trigger once for <u>each</u> record, once for <u>all</u> records, or <u>once per rep</u> who has high priority calls.

8.2.3.2 Event Details

For each checked event, you will see the event's application, name, date/time of execution, and runtime (processor seconds) that the event took to complete.

For each <u>triggered</u> event, you will see the above details plus a blue plus sign that enables you to see the email alert message text, the "deliverables" of the event (its alerts, reports, et cetera), and the value of the unique IDs of the triggered records that have met the event's criteria.

For an event that ends in error, you will see the event's application, name, date/time of execution, as well as the number and description of the error that occurred.

8.2.3.3 Displaying Email Alert Message Text

When on the "Triggered" tab, you may display the full text of the email alert message. To do so, click on the blue plus sign next to an event and then view the tab called "Email Message Text". Note that this tab will not appear if an event is not configured to send alerts via email.

An event's mail message text will appear such as the following:

Event Details

The following customers have invoices that are overdue for payment

Customer No.	Customer Name	Primary Contact	Phone	Invoice No.	Invoice Date
01-BRESLIN	Breslin Parts Supply	Winnie Spackman	(414) 555-9654	MAR3190	3/31/2020 12:00:00 AM
01-HILLSB	Hillsboro Service Center	Pam Wampler	(414) 555-6599	FEB2890	2/28/2020 12:00:00 AM
01-HILLSB	Hillsboro Service Center	Pam Wampler	(414) 555-6599	JAN3190	1/31/2020 12:00:00 AM
01-HILLSB	Hillsboro Service Center	Pam Wampler	(414) 555-6599	MAR3190	3/31/2020 12:00:00 AM

DISPLAYING ALERT MESSAGE TEXT

8.2.3.4 Displaying Event Deliverables

Also on the "Triggered" tab, you may display an event's "deliverables" – all of the content an event is configured to generate (such as alerts and reports) as well as all of an event's workflow actions – such as triggering updates into one or more software applications. You'll see a window like the following:
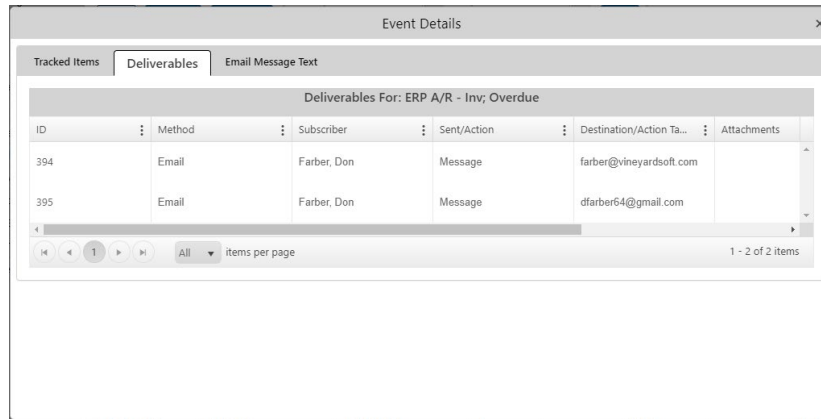
DISPLAYING AN EVENT'S DELIVERABLES

8.2.3.5   Displaying Event's Tracked (Triggered) Items

Lastly on the "Triggered" tab, you may display an event's "tracked items" – the unique record identifications of each record that has met an event's criteria when the event was triggered. This will look like the following:
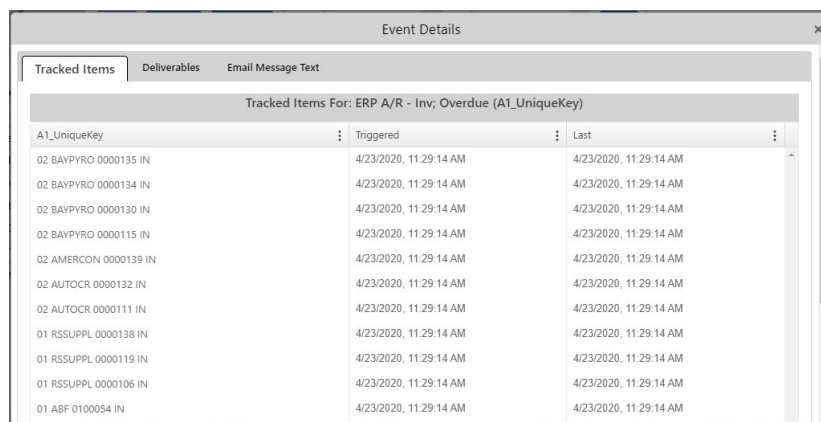
DISPLAYING AN EVENT'S TRACKED/TRIGGERED RECORDS

> *A record that meets an event's criteria will remain in a triggered state until (1) the record no longer meets the event's criteria, and (2) someone has "cleared" the triggered record for that item.*

You will <u>not</u> have access to an event's tracked items if:

- The event's trigger does not have a "unique" column identified, or;
- The event is configured to "repeat notifications for triggered events"

8.2.3.6   How to Handle Event Errors

When an event fails because of an error, three things happen:

- The event will appear in the "Errors" branch.
- The event will appear in the "Pending" branch.
- KnowledgeSync will <u>**re-submit**</u> the event 'x' times (a KnowledgeSync system setting) until it either runs successfully, is deleted, or is manually completed.

KnowledgeSync retains only one iteration of an event that errors. If an event errors due to a condition that corrects itself (such as a server down), you can leave the event alone as once the condition is corrected, the event will successfully execute.

If the event errors because of an incorrect configuration, you can edit the event, correct it, and your "fixed" version will be used the next time the event runs. If the event errors and you wish to remove that run of the event, a KnowledgeSync user with the appropriate authorizations may either delete or manually complete the record from within the Monitor option in which the error appears.

8.2.4   Email, SMS, File (FTP/SFTP) & Webcast Delivery

The "Delivery" folders for email, SMS, file and webcast (dashboard) are divided into three branches:

- **Pending**. A "pending" item is one that has not been sent either because it is currently processing/awaiting processing, or because of the item has encountered an error.
- **Sent**. A "sent" item is one that was successfully delivered.
- **Error**. A "error" item will appear if the item could not be successfully delivered. An erroring item will automatically be placed back into the "Pending" branch.
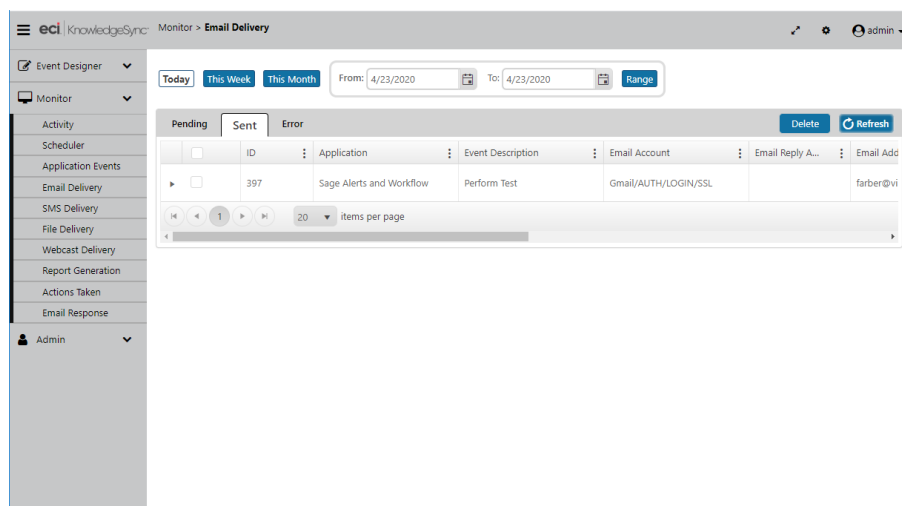
EMAIL "SENT" DELIVERY FOLDER

8.2.4.1   How to Handle Message Delivery Errors

When a message fails to be delivered successfully, three things happen:

- The message will appear in the "Errors" branch
- The message will appear in the "Pending" branch
- KnowledgeSync will attempt to re-send the message 'x' times (determined by a system setting) until the re-send is either successfully delivered or is deleted or manually completed.

8.2.5   Report Generation

The "Report Generation" branch lets you track the status of forms, documents, and reports that an event is generating. This folder is divided into three branches:

- **Pending**. A pending report is one that has not been generated either because it is currently processing/awaiting processing, or because of the report has encountered an error.
- **Generated**. A "generated" report is one that was successfully produced. Just because a report was "generated", it does not mean it was successfully delivered. Review the appropriate delivery method (such as email) to see if the report was delivered to its intended recipients.

- **Error**. A report will show up under the errors branch if it failed to be generated successfully. A report that fails will automatically be placed back into the "Pending" branch.
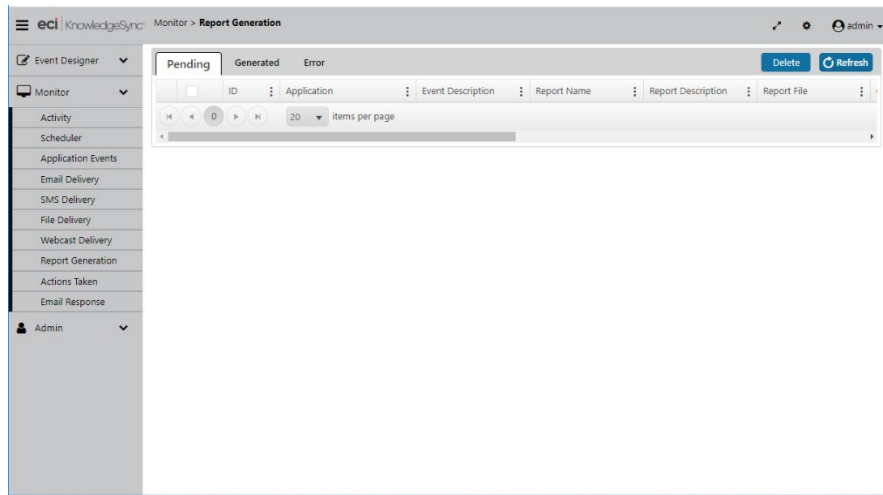
REPORT GENERATION FOLDER

8.2.5.1 How to Handle Report Generation Errors

Report generation errors can occur for several reasons, typically an error in the report's design. When a report fails to be generated successfully, three things happen:

- The report will appear in the "Errors" branch
- The report will be re-queued under the "Pending" branch.
- KnowledgeSync will attempt to re-generate the report 'x' times (determined by your administrator) until it either succeeds or the report is deleted or manually completed.

8.2.6 Actions Taken (Workflow)

"Actions Taken" refers to events that execute workflow actions in response to a triggered event. This folder is divided into three branches:

- **Pending**. A pending action is one that has not been completed either because it is currently processing/awaiting processing, or because of the item has encountered an error.
- **Completed**. A "completed" action successfully executed.
- **Errors**. An action will show up under "errors" if it was unable to successfully execute. An action that fails will automatically be placed back into the "Pending" branch.

8.2.6.1 How to Handle Workflow Action Errors

Action errors occur for a number of reasons, such as a syntactical error in a web service call or within a VB script.

When a workflow action fails, three things happen:

- The action will appear in the "Errors" branch
- The action will appear in the "Pending" branch
- KnowledgeSync will attempt to re-execute the action 'x' times until it either succeeds or is manually deleted or completed.

8.2.7 Email Response

(See chapter titled Email Response System for more details.)

The Email Response folder lets you see what e-mail accounts are being checked for incoming messages, how many messages have been received and loaded into KnowledgeSync, and if any errors were encountered.

Once email message are loaded into KnowledgeSync, the corresponding events are treated just like any other application event, and thus can be tracked using the "Application Events" folder.

The "Email Response" folder is divided into three branches:

- **Checked**. Incoming email messages that have been checked by KnowledgeSync.
- **Received**. Incoming messages that have been loaded into the KnowledgeSync email response system database.
- **Errors**. Errors that KnowledgeSync encountered while trying to check for messages sent to one or more email accounts.
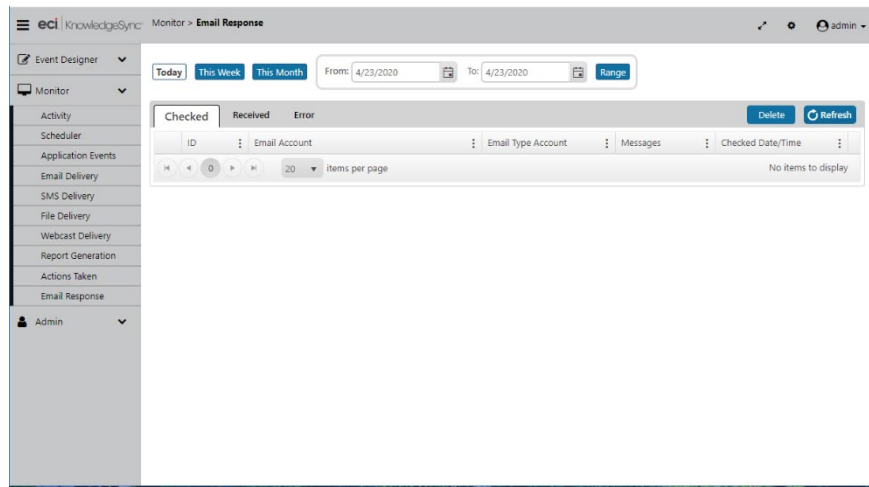
EMAIL RESPONSE SYSTEM ACTIVITY IN THE MONITOR

9 The Email Response System (ERS)

The Email Response System enables KnowledgeSync to monitor (and respond to) the content of incoming email messages in much the same way that you can monitor the contents of application databases.

If an email message meets certain criteria, you can:

- Re-route the message to specific recipients
- Notify people about the message
- Auto-reply to the sender
- Trigger the generation and delivery of specific reports and/or forms
- Match the sender of the email with a record in an underlying database
- Use the contents of the message to add or update information in an underlying database

ERS refers specifically to the ability to monitor **incoming** messages and respond to them; if your interest is only in sending **outgoing** alert messages via email, you do not need to use the ERS.

Authentication and secure sockets (SSL) are supported in ERS.

9.1 ERS and Web Forms

ERS can monitor email messages coming from any source; this includes messages generated from data entered on a web form, as web forms typically create email messages containing data from the completed form. Thus with the ERS, you can take the contents of email messages created from a web form and then use KnowledgeSync's standard capabilities to trigger application updates, send alerts, auto-respond to the person who filled out the form, and so on.

9.2 Supported Email Systems

ERS supports all of email systems listed in the KnowledgeSync Email Account Setup option documented earlier in this manual (under KnowledgeSync Administration).

9.3 How ERS Works

Here is an overview of how ERS works:

- You identify what email accounts KnowledgeSync will monitor for incoming messages.
- When a new message arrives, KnowledgeSync loads its data into an internal table called "Inbox".
- Queries check the content of the incoming email message to see if the message meets certain criteria.
- Events that use these queries send out alerts, auto-respond to the sender, and (if appropriate), and update application databases with the message data.

The key to this process is that KnowledgeSync takes each incoming message and creates a database record (in the "inbox" table) out of it. As a result, standard trigger (query) and event design functions are used to process and respond to each message.

9.4 What This Chapter Covers

This chapter covers the steps to configure KnowledgeSync to monitor incoming email and load those messages as records into the table called Inbox.

Once those email records are loaded into the "inbox", you may refer to the chapters on "Query Designing" and "Event Designing" for details on how to identify and respond to those email message records.

9.5 Create Connection for ERS Database

KnowledgeSync processes incoming email messages by loading them into a table called Inbox which is located in the SQL Server database called KS_Inbox. Connecting KnowledgeSync to this underlying SQL Server database may be done via any of the methods detailed in the Query Designing chapter of this guide (.NET database provider, OLE/DB, or ODBC).

Optimum system performance is obtained by using the SQL .NET data provider.

9.6   Identify Email Accounts to Monitor

To identify the email accounts that will be monitored for incoming messages, follow these steps:

Expand the Administrator option, select "Services" and then "Email Accounts".
If the account you wish to monitor for incoming message is listed, click on the blue Edit button; if not, click on the "Add" button.

9.6.1   Email Account Properties

KNOWLEDGESYNC EMAIL ACCOUNT SETUP WINDOW

There are five options for Automatic Email Setup:

- Gmail
- Hotmail
- Yahoo mail
- Exchange (non-hosted)
- Outlook 365 (hosted Exchange)

If you will be using one of these methods, click on the corresponding button at the bottom of the email setup window. Doing so will auto-fill the settings for these email systems. All that is needed is the login (username), password, and the SMTP Send from Name. Please do not change any other values.

Please refer to the following Microsoft Article for further information on Outlook 365 account settings:

http://office.microsoft.com/en-ca/outlook-help/settings-for-pop-and-imap-access-for-office-365-for-business-or-microsoft-exchange-accounts-HA102908389.aspx

In the following example the Outlook 365 option was chosen. You can see the fields that require information input as they contain the word "your", (i.e. youraccountname, yourserver.outlook.com).

OUTLOOK 365 EMAIL ACCOUNT SETUP

The following describes the email account configuration fields:

- **Email Type**: Select from Internet, MAPI, Exchange, or VIM. We recommend INTERNET.
- **Account Description**: This value defaults based on which button at the bottom of the email setup window was selected. Do not modify this value unless instructed by KnowledgeSync Support personnel.
- **Login Name**: Fill in this value based on the default format that appears.
- **Password**: The email password for this user.
- **Active**: Whether this account is currently available for sending email (you can configure an account without enabling it for immediate use).
- **TLS**: Whether TLS authentication and encryption is enabled for this account.
- **Authentication Method**: Select from SSL, LOGIN, NTLM, MSN, CRAM-MD5 and AUTO.
- **Outgoing mail SMTP Server Name**: The server that will be used when sending email notifications via the INTERNET method.
- **SMTP From Name**: The name that will appear as the "sender" of the email alert message. It is required when using type INTERNET and it MUST be a valid email address with a valid domain name. For example, a valid from name will look like the following: **KnowledgeSync <youraccountname@gmail.com>**
- **SMTP Port**: The port that will be used to accommodate firewall security software. Leave this field blank unless a default value appears.
- **Incoming Mail Server Name**: The name of the server that monitors the inbound email.
- **IMAP Port**: The port that will be used to accommodate firewall security software. Leave this field blank unless a default value appears.

9.6.2   Email Response System Options

When you click on the "Email Response System Options (Incoming)" tab, the following window opens:
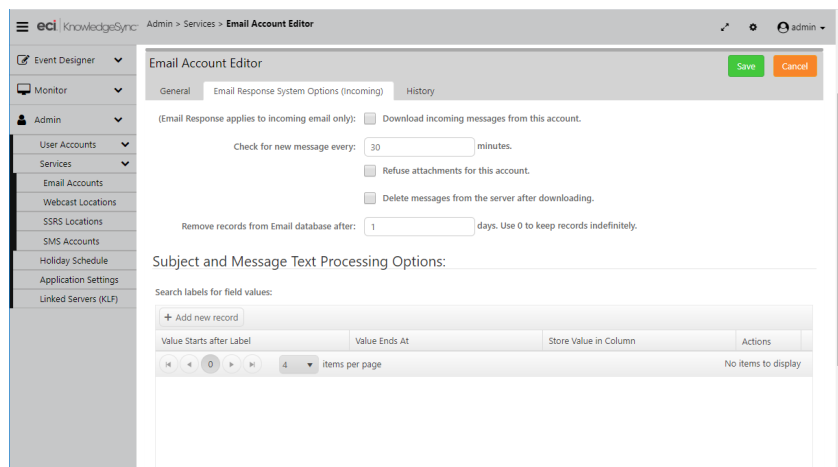
EMAIL RESPONSE SYSTEM OPTIONS (INCOMING)

The following is a brief description of the ERS configuration options:

- **Download incoming messages**. Check this box if KnowledgeSync should download messages sent to this account.
- **Check for new messages every 'n' minutes**: Specify how often (in minutes) KnowledgeSync will check for new messages sent to this account.
- **Refuse attachments for this account**: Check this box if you do <u>not</u> want to download email attachments. (ERS cannot scan the contents of attached files but can scan the <u>names</u> of attachments.)
- **Delete messages after downloading**: Check this box to delete downloaded messages from the email server after they are stored in the ERS database. Check this option <u>only</u> if there are no other applications or email clients that need these messages. KS moves the messages to the **Deleted Items** folder.
- **Remove records from database after 'n' days.**: Specify the number of days that email records will be kept in the ERS database. KnowledgeSync recommends 120 days; specify a value of '0' to keep email records indefinitely.

9.6.2.1   ERS & Attached Files

Although the ERS cannot scan the <u>contents</u> of attached files, it can identify whether a message has attachments, and the names of them. Additionally, attachments are automatically stored in the KnowledgeSync "Attachments" folder, and an ERS event may copy or move the attached files to a location of your choice.

Attachment files are always created with unique names.

9.6.2.2   Message Text Parsing

KnowledgeSync can parse through the contents of incoming messages and can load individual elements of message data into specific fields in the ERS inbox table.

For example, consider mail message text like the following:

Company Name: Acme Corporation
Address: 1 Main Street
Contact Name: Robert Smith
Phone: 800-555-1212
Request: Please send me information you have regarding your product.

You can tell KnowledgeSync to take the value after "Company Name:" and store it in the inbox field called "account". Likewise, the value after "Address:" could be stored in an Inbox field called "address", and so on.

This is referred to as "parsing" the data in an incoming message and mapping to a corresponding field in the inbox table. The inbox table comes pre-configured with a number of fields that can have data parsed into them from an incoming message; you may also modify the inbox table to include additional fields that you'd like to map data to.

9.6.2.3   The Inbox Table
You can modify the Inbox table to add columns or modify the names of existing columns. You must stop the KnowledgeSync service before making changes and re-start the Service when done.

9.6.3   Pre-Defined Loadable Inbox Fields
The following fields in the inbox table may be loaded with data from incoming messages.

Inbox Field Name	Default Purpose
Account	The sender's account name
Account_no	The sender's account number
Address	The sender's address
Application	The application the sender is writing about
Case_no	The case number referenced in the message
City	The sender's city
Country	The sender's country
Email	The sender's preferred reply-to email address
Fax	The sender's fax number
First_Name	The sender's first name
Full_Name	The sender's full name
Incident_No	The incident number referenced in the message
Last_Name	The sender's last name
License_No	The license number referenced in the message
Mobile	The sender's cell phone number
Notes	Message notes
Order_No	The order number referenced in the message
Pager	The sender's pager number
Phone_Home	The sender's home phone number
Phone_Work	The sender's work phone number
Postal	The sender's postal (zip) code
Problem	The description of the problem/issue
Product	The product that the sender is writing about
Question	The question text
Request	The request details
Serial_No	The serial number
Service	The description of the service required
Service_No	The service number referenced in the message
Solution	The solution text for a problem or question
State	The sender's state
Ticket_No	The ticket number referenced in the message
Title	The sender's title
Tracking_No	The tracking number referenced in the message

Udf_date1- Udf_date3	Date and time-based user-definable fields.
Udf_Flag	Integer-based user-definable field.
Udf_Memo	Memo-based (long text) user-definable field.
Udf_num1- Udf_num3	Real number-based user-definable fields.
Udf1 – Ufld5	Five user-definable ERS data fields
User_Name	The sender's username.
Version	The version of the product referenced in the message
Web_Site	The sender's web site address

9.6.4 How To Specify Inbox Parsing

The order in which you specify the parsing fields is very important; the order that you designate in KnowledgeSync must be the same as the order that the fields appear within incoming email messages. To configure mail message parsing, go to the bottom of the Email Response System (Incoming)" tab – where it says "Subject and Message Text Processing Options" and follow these steps:

Click on the "+ Add New Record" button.
Key in the text that will precede the value (e.g., "Account:").
Click in "Value Ends at" and choose from:
- o Carriage return
- o Line feed
- o Blank space
- o End of message subject
- o End of message body
- o End of subject or message

Click in "Store Value in Column" and select (from the drop-down list) the KS Inbox table's column to load this value into (e.g., "account_no").
Repeat for each field you wish to map.
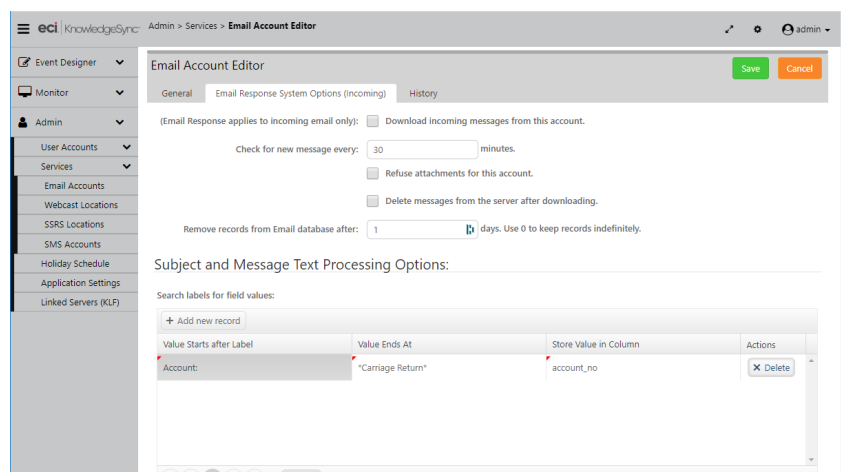
INBOX MESSAGE TEXT PARSING

9.6.5 Message Posting Options

At the bottom of the Email Response System tab, you have the option to specify whether you would like to post (store) downloaded mail messages in another database for historical retention. To use this function, follow these steps:

Check the option to "Post incoming messages to an alternate database".
Choose the database connection (e.g., provider or ODBC source) that points to this database.
Enter the name of the table (within the destination database) to which the inbox records will be posted.

The destination database table must be configured to exactly mirror the column names that exist within the inbox table, and the alternate database table must have a unique primary key.

9.7 Create ERS Connection (Outside of KnowledgeSync)

ERS data is stored in a SQL Server database called "KS_Inbox". As detailed earlier in this chapter you'll need to create a connection to that database using either a .NET data provider (preferred connection method), ODBC, or OLE/DB in order to allow KS to monitor and respond to incoming emails.

9.8 Create ERS Application (Inside KnowledgeSync)

ERS queries and events must be stored within their own application in KnowledgeSync. To configure this, follow these steps:

Log into KnowledgeSync and expand your list of application events. If a branch for the Email Response System exists, click on it and view its "Properties". If an ERS branch does not exist, click on the "New App" button and create an application called "Email Response System" or simply "ERS".
Select the connection that points at the inbox database and be sure to supply the required credentials.
Save and close your work.

9.9 Inbox Queries

ERS queries retrieve data from the inbox table. This table contains both "loadable" fields (detailed earlier in this chapter) as well as pre-loaded fields that KnowledgeSync populates with details from incoming messages.

The following list describes all the inbox pre-loaded fields. A query may retrieve and use data from any combination of loadable and pre-loaded data fields. Please do not modify or remove any of the following fields from the Inbox database.

Inbox Field Name	Purpose
Attachment_Count	Number of attachments including alternate HTML text
Attachment_Files	Comma separated list of attachment file specifications. Attachments are stored in the attachments folder.
Attachment_Names	Comma separated names of the attachments
Attachment_Start	(0, 1, or 2) '0' indicates no attachments, '1' indicates one or more attachments, '2' indicates attachments beyond alternate HTML text.
CC_Names	The "cc" names from an incoming message
Certified	Whether the message was sent certified.
Created_dt	The date/time the mail message was created in the inbox
Email_Account	The name of the email account the message came in to.
Email_Account_Id	The ID email account this message came in to.
From_Name	The email "from name", such as 'John Smith'
From_Path	The full path of the sender's name, such as 'John Smith <smith@mycompany.com>'
From_Root	The email address of the sender, such as 'smith@mycompany.com'
ID	ERS-assigned unique column
Inet_Alt_Count	The number of "alternate" attachments (for Internet type mail systems only.)
Inet_Mail_Program	The name of the email client used to send the email (for Internet type mail systems only)

Inet_Message_Length	The length of the message (in bytes) (for Internet type mail systems only)
Inet_Priority	The priority of the message (for Internet type mail systems only)
Inet_Raw_Header	The header of the email message (for Internet type mail systems only)
Inet_UIDL	Internal Internet mail identifier (for Internet type mail systems only)
Inet_Webform	'0' (not from web form) or '-1-' (from web form) (for Internet type mail systems only)
Inet_Webform_Fields	The names of the web form's fields (for Internet type mail systems only)
Inet_Webform_Values	The values of the web form's fields. (for Internet type mail systems only)
Login_Name	The login name of the account for this message
Message	The mail message text
Message_Id	The unique message id (for Internet type mail systems only)
Received	The date the message was received
Received_dt	The date and time the message was received
Sent_dt	The date and time the message was sent
Subject	The subject of the message
To_Names	Whom the email was sent to.

9.9.1   Inbox Query Filters

Creating filters for inbox queries is done in the same manner as creating filters for application events. (See chapter on Query Designing)

Specific to querying incoming email records, however, is the need to test for messages that are sent from specific email addresses or to specific email addresses.

The following filter identifies any messages sent from a person at KnowledgeSync:

```
Inbox.from_path    like    %KnowledgeSync.com%
```

This next filter identifies any messages sent to KnowledgeSync's support mailbox:

```
Inbox.to_names    like    %support@KnowledgeSync.com%
```

You should also limit query results to only those messages received today:

```
Inbox.received   is equal to   {%Current Date%}
```

9.10 Inbox Events

Inbox events are created just like standard application events. However, you might decide that you'd like to link an incoming email message to a corresponding contact record within a customer database. This would let you:

Identify the corresponding account manager and alert them
Update the customer's record with the details of the incoming message
Create a record (such as a support ticket) and link that ticket to the customer
Identify if the sender is not already in your customer database

9.11 Linking Mail Messages to Client Records

You can link an incoming email message to a customer record in two ways:

Using two queries in one event
Using a Visual Basic script or a web service call

Although VB scripts and web service calls give you complete control and flexibility in terms of creating database records based on receipt of an incoming email, both methods require programming expertise. The use of two queries requires no programming knowledge.

When you use two queries in an inbox event, the first query retrieves the email records, and the second query links those records to your customer database.

This requires identifying a field in the inbox record that matches with a piece of data from the customer record. This might be the sender's email address (stored in the "from_root" column in the Inbox table), but could also be any other field in the incoming message that corresponds to a matching field in your customer database (such as a customer's "account number").

Using a field such as the sender's email address to match to a customer's record in an application database, your two queries would work as follows:

Query #1 (retrieving inbox records) would include the linking field (i.e., the sender's address, "from_root") as one of the fields in the query.
Query #2 (retrieving customer info) would have a filter that compares the email address in the customer application (e.g., "customer.mail") to the customized name of the inbox sender address field (e.g., "{from_root}").

This is illustrated in the following:
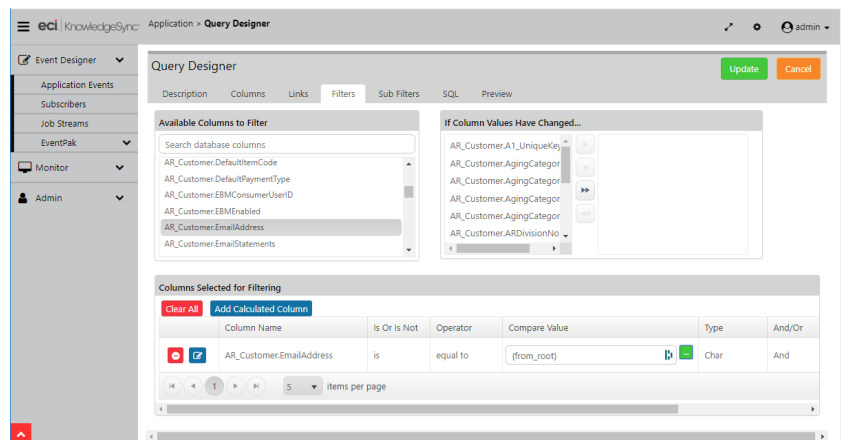
MATCHING EMAIL ADDRESS TO DATABASE RECORD

An event that uses two queries in this manner will trigger only if the sender's address is also found in the customer database. Once a match is found, all of the fields of data from both queries may be used in alert messages and in workflow actions.

(And, although the above scenario identifies message senders who exist in a customer database, you could construct a similar event where your second query looks for no matches to your customer database. This would identify a new client or prospect.)

9.12 Alerts & Actions

In response to an inbox event, KnowledgeSync can send messages to:

The salesperson and/or support person associated with an account
The sender of the incoming mail message
Other people related to the incoming message (such as "cc" names)

These messages can include all standard content associated with a database event. Here is an example of a message that could send back to a person who logged a support question via email:

Hello {from_name},

This is to confirm our receipt of your email sent on {sent_dt} to {to_names}.
We will contact you shortly when we have an answer for you.

Thank You,
Support Team

KnowledgeSync can also execute workflow actions because of a triggered inbox event. One common use of an inbox action is to have an event take information from an incoming email and pass that into a VB script or web service call to create or update records within an underlying application database.

9.13 Using Email for Report Requests

You can use ERS to allow users, partners, or clients to request and receive forms, documents, or reports. Consider the scenario where an organization has staff who need to need to run reports, but currently must ask their IT department to run the reports for them.

Using the ERS module, an organization could configure a mailbox called "report_requests" and whenever a staff member needs a report, they could send a message to that account. The message could include the name of the report to run and even the selection criteria for that report.

When such a message is emailed, KnowledgeSync receives it, uses a query to determine which report is being requested, and then runs the report and emails it back to the requester. All done automatically, without human intervention.

9.14 Inbox Event Schedule

Since you want KnowledgeSync to respond to incoming mail messages as soon as they are received, all inbox events should use a frequency of "When Email Arrives."

9.15 Inbox Events in the Monitor

To monitor the status of inbox events, use the Monitor options. Although inbox events turn into standard application events, there is an additional branch called Email Response.

The Email Response branch lets you check to see what mail messages have been received (and logged) into the inbox database.

Once an incoming message has been received and loaded into the inbox database, you may use the standard application event and delivery branches of the Monitor to check to see if the inbox event was triggered, who was notified, what applications were updated, and so on.

10 Generating & Delivering Reports, Forms & Documents

KnowledgeSync includes the ability to trigger, generate and deliver reports (which includes standard forms and documents designed using report templates) for both Crystal Reports and Microsoft SQL Server Reports (SSRS).

KnowledgeSync can generate and deliver reports:

On a scheduled basis, such as daily at 9 AM or every Friday at 5 PM.
When an event's conditions are met (referred to as a triggered report), such as generating an A/R aging report for a client when that client gets to within 10% of their credit limit or generating an invoice when a new order is placed.

You do not need to install either Crystal or SSRS on the KnowledgeSync server but do check to see if the KnowledgeSync server already contains a version of Crystal Reports on it. You can have only a single version of Crystal Reports on a server and the most recent installation of Crystal will overwrite any previous installations.

10.1 Supported Report Types

Please note the following about KnowledgeSync's Support of reports:

All reports must be connected to the underlying database via .NET data provider, OLE/DB, or ODBC.
Crystal reports must be based off of the permanent data tables in a database. I.e., reports cannot make use of "temp" or "work" tables. (If your reports do use temp or worktables, you'll need to make a copy of them and then modify those copies to point at an application's permanent data tables & views.)
Crystal reports cannot make use of *.ttx files or "print parameters,"and cannot make use of programmatically defined parameters. All Crystal report parameters must show up in the "Reports" tab of a KnowledgeSync event under the heading of "Report Parameters".
KnowledgeSync can use an event's "actions" to generate and deliver other (non-Crystal, non-SSRS) reports, such as Sage Intelligence reports. Contact KnowledgeSync Support for details.

10.2 Reports vs Forms & Documents

KnowledgeSync is frequently used to create and deliver forms and documents, such as order confirmations, invoices, statements, dunning notices, picking lists, quotes, purchase orders and other similar materials.

KnowledgeSync can generate and deliver such documents using a report as the template for each form.

10.3 Supported Versions of Crystal & SSRS Reports

Please refer to the KnowledgeSync Installation Guide for details regarding Crystal Reports compatibility. KnowledgeSync has been upgraded to support Crystal runtime versions 13.0.32 and 13.0.33. As with previous versions, KnowledgeSync V10.6 detects the Crystal Reports Runtime version on your machine and uses it to generate Crystal Reports. If the ERP or other product being used with KnowledgeSync needs a Crystal runtime version 13.0.32 or higher, KnowledgeSync will need to be uninstalled and re-installed after the ERP product has been upgraded.

KnowledgeSync supports SSRS versions for MS SQL 2012 and onwards (e.g., SQL 2022).

10.4 SSRS Location Definition

For SSRS reports, you need to specify the location of your SSRS installation. Expand the KnowledgeSync "Admin" branch and under "Services", click on "SSRS Locations." When you add or update an SSRS location, the following window will appear:

SQL SERVER REPORTING SERVICES (SSRS) SETUP

Fill in the fields as follows:

Description: The name of the SSRS server (e.g., "SSRS on QA-SVR123")
Report Server URL: This is the URL of the SSRS Report Execution 2005 Service on the report server (e.g., "http://qa-svr123/ReportServer"). Copy this URL from SQL Server Reporting Services Configuration Manager.

> *Make sure the URL is accessible from the machine where KnowledgeSync (the web application and Windows Service) are running. Use a "localhost" URL only if KnowledgeSync and SSRS are installed and running on the same machine.*

Report Server 2010 URL: This is the URL of the Report Service 2010 Service on the report server.
Notes: Use this field to contain comments about the corresponding SSRS location.
Username/Password/Domain: This is where you specify the credentials used to run the Report Execution service of SSRS. The values in these fields must match the settings in the SQL Server Reporting Services Configuration Manager.

> *The domain can be entered separately (in the "Domain" field) or it may be included in the username, as in "mgmt\svcQASQL".*

10.5 Report Definition
Once you have defined your SSRS "location" (see previous section), the next step in enabling reports is to tell KnowledgeSync which Crystal and/or SSRS reports you wish to generate. Reports are organized according to the application they work with, and the reports must already have been created in Crystal or SSRS in order to define them within KnowledgeSync.

> *If you are using Crystal reports, the option to "Save Data With Report" must underline not selected from within Crystal and the reports defined in KnowledgeSync underline must (in Crystal) be configured to use the same connection (data provider) as the corresponding application in KnowledgeSync.*

To define your reports, log into KnowledgeSync open the Application Events branch, expand the application in which you wish to define reports, and click on the "Reports" branch.

Click on the **New Report** button to define a report; the following window opens:

Report Definition		Update	Cancel

Report Type: *	Crystal ▼
Report Name: *	test
Report Description:	test
Report Path: *	C:\testing\EmployeesV4_DiffLinkedParams.rpt
Output File Name (Optional):	
Separate Connections for Sub-reports:	✓

Sub Report	Connection
EmployeePayRates	
EmployeePersonalData	

Created By:	admin
Created On:	10/27/2020, 8:06:29 PM
Last Changed By:	ADMIN
Last Changed On:	4/26/2022, 12:41:11 PM

NEW REPORT DEFINITION

Fill in the fields as follows:

- **Report Type**: Select either Crystal or SSRS. Select if the type is On Prem or Hosted.
- **Report Name**. A short description of the report.
- **Report Description**. Use this field to contain a detailed description of the report.
- **Report Path**. This is where you specify the location of the source file that corresponds to the report you are adding. If the report is a Crystal one, the selected file will end with *.rpt - if the report is an SSRS one, you will need to specify the path of the report as deployed on your SSRS report server

  Do not reference mapped drives; use UNC format instead. If you refer to a destination on another computer, make sure the user account running the KnowledgeSync service has read rights on the other computer.
- **Report Output Filename**. The name that will be given to the output of this report. Do not specify a disk or directory; the file will be created in the KnowledgeSync "Reports" directory.

  If blank, the name will be generated by KnowledgeSync. If you are hard-coding the output file name, the name must not contain any non-standard characters, such as @, #, $, %, ^, &, *, (,), _, and +.

  Do not specify the file suffix (e.g., .pdf, .html, etc.). KnowledgeSync will append that automatically, based on the selected output format of the report.

  If the report is generated multiple times by the same event, do not hardcode the output name as this will cause each generation to overwrite the last.

- **Separate Connections for Sub-reports**. Reports and sub-reports can connect to more than one data source and different databases. Check this box if the reports are sent to multiple connections. Select a connection for each Sub-report.

10.5.1 Dynamically Naming the Report Output File

You can have KnowledgeSync **dynamically** name the report output file, based on data from the triggered records. For example, an event that triggers an invoice (using a report as the invoice template) could name each generated invoice using a combination of the client's name and order number.

To do this, select from the names of the trigger fields (e.g., a query's columns) you wish to use in your report output files, Thus – using the preceding scenario – as long as you had selected the client name and order number in the trigger that is used by the event generating the invoices, you could specify an output file name of:

```
{cus_name}_{ord_no)
```

Note that you will need to refer to an associated trigger (and/or to KnowledgeSync's list of date substitution variables) to get the names of any variable fields you wish to use in the report output file specification.

Repeat this process for each report you wish KnowledgeSync to generate.

10.6 Scheduled vs. Triggered Reports

There are two kinds of report events you can create in KnowledgeSync:

Scheduled. A scheduled report event is an event that runs on a recurring schedule; such an event has one or more associated reports, and <u>no triggers</u>. Examples of scheduled reports are:

- o   Daily Sales report
- o   Weekly shipping summary
- o   Monthly trend analysis

Triggered. A triggered report event is an event that uses one or more triggers to see if certain data conditions exist; if the specified conditions do exist, the event is triggered and it generates its associated report(s). Examples of triggered reports are:

- o   If a client's balance exceeds $10,000, generate an A/R Aging report for that client.
- o   When a new order is placed, deliver an order confirmation (report) to the client.
- o   If an inventory item falls below its re-order point, generate a stock status report.

A single scheduled report event may generate and distribute multiple Crystal and/or SSRS reports.

Both scheduled and triggered reports are configured as <u>events</u> in the KnowledgeSync system.

10.7 Preparation for a Report Event

If you're creating a <u>scheduled</u> report event and you've already defined the reports you wish KnowledgeSync to generate, you can proceed directly to the option to create a new event. (See "Scheduled Reports".)

If, however, you wish to create an event that checks for certain data conditions and then (if those conditions are met) <u>triggers</u> an associated report, then you'll need to begin by designing the trigger for your report event.

The trigger in a report event performs three functions:

It determines whether the event's conditions are met and thus whether the report associated with such an event gets generated and delivered.
It provides access to the data fields that can be passed into the report as selection criteria. (E.g., using the example of a report triggered by clients whose balance exceeds $10,000, the trigger would pass the ID of each matching client into the report.)

It supplies the "delivery address" of the person(s) to receive the generated reports. (E.g., in the previous example, the trigger could retrieve the email address of the salesperson assigned to any client whose balance is greater than $10,000 and thus deliver each report to the corresponding salesperson.

10.8 Scheduled Reports

To create a scheduled report event – that is, an event with no trigger and with one or more associated reports, follow these steps:

Expand the application you are working with, click on the "Events" branch and click on "New Event". Fill in the fields much as you would a standard KnowledgeSync event (see the Event Designing chapter for details).

Be sure to check the "active", "include reports", and "keep only last checked record" boxes. Your window will look like the following:
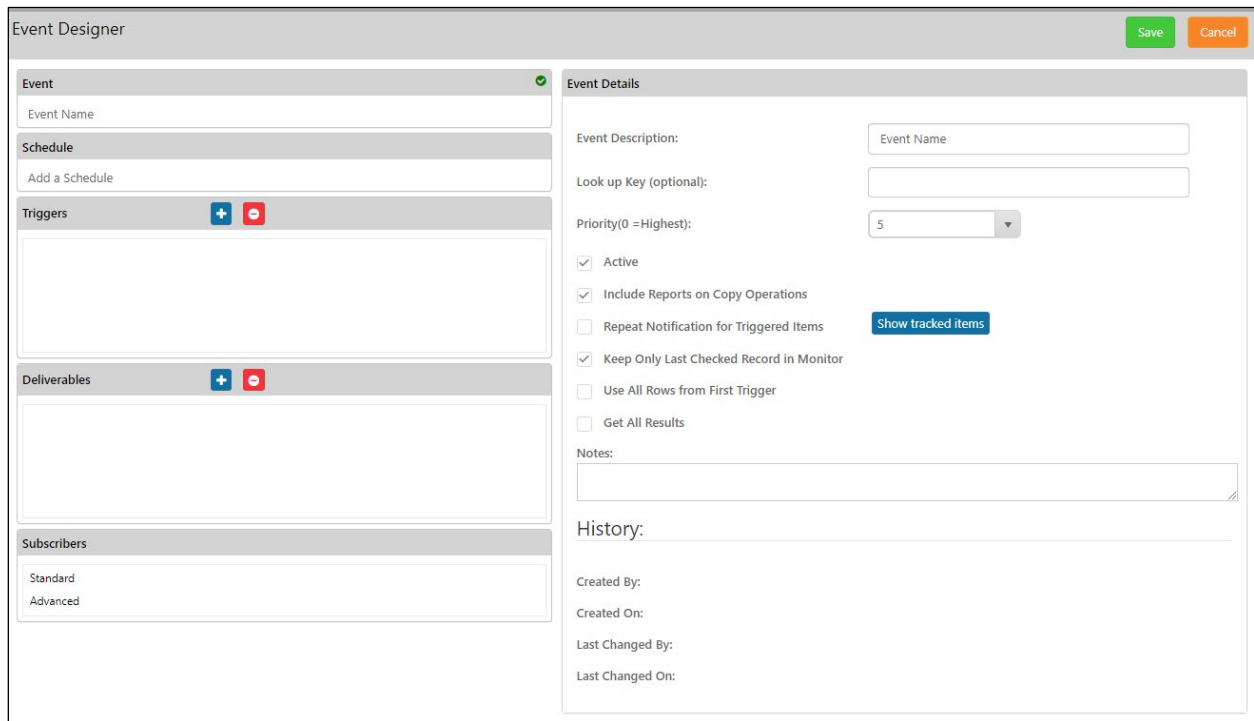
SCHEDULED REPORT EVENT

10.8.1 The Report's Schedule

A scheduled report (as opposed to a triggered report) runs on a recurring schedule, such as daily, weekly, or monthly. Click on the description beneath the Schedule heading and select the schedule that controls how often this event is generated and delivered.
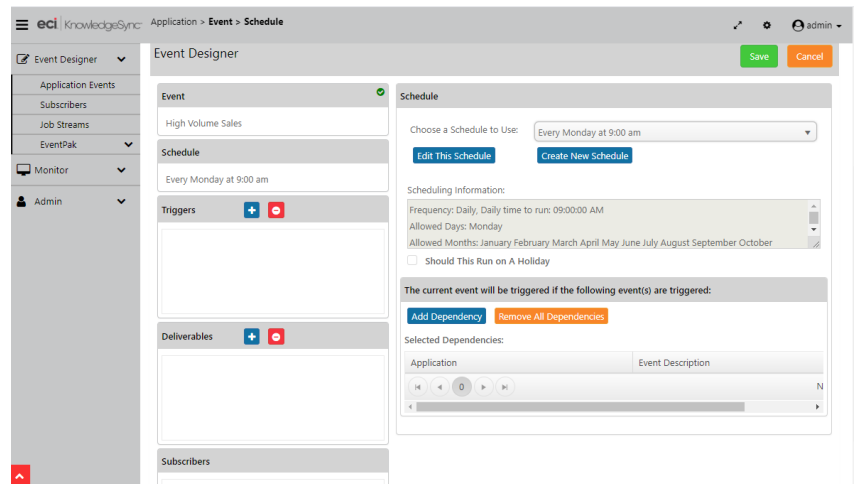
THE REPORT'S SCHEDULE

10.8.2 Do Not Specify Any Triggers

Since a scheduled report runs on a recurring basis (and is not triggered by data conditions), be sure to have no triggers defined for a scheduled report event.

10.8.3 The Event's Report(s)

A single scheduled report event may produce one report (e.g., "Weekly Sales Report"), or it may produce multiple reports (e.g., "Month-End Performance Reports"); you can have up to 999 reports associated with a single event. To add a report to an event, follow these steps:

Click on the blue Plus sign next to "Deliverables".
Under "Alert Content", click on the "Report" button. The following window opens:
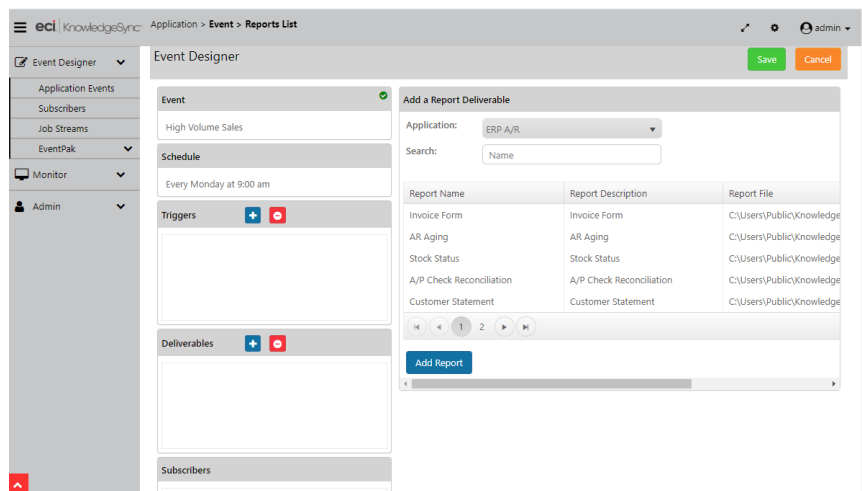
AVAILABLE REPORTS FOR AN EVENT

By default, you will be shown a list of the reports that are defined for the current application; select a different application if your desired report was defined for an application other than the one you're currently in. (You may also use the "Search" window to locate a report by its name.)
Once you have located the desired report, single-click on it then click on "Add Report".

10.8.3.1 The Report's Output Type (Style)

(If you do not see the "Report Properties" window on the right-side of your display, click on the report you just added to the event – beneath the "Deliverables" heading.)

On the Reports Properties window, click in the Report Type column to choose the report output format

- MS Word
- MS Excel (.xlsx)
- MS Excel legacy format (.xls)
- PDF
- HTML
- MHTML
- Comma Delimited
- Tab Delimited
- XML
- Image
- Ascii text
- Rich Text
- Crystal

Notes: SAP Crystal Reports runtime only allows data-only exports to .xlsx files (formatting is not exported).

The output option of Crystal creates a .rpt file that may be viewed only by users who have the Crystal client installed on their desktop.

10.8.3.2 The Report's Properties

If you do not see the Report Properties window on the right-side of the Event Designer display, click on the report you just added to the event – beneath the Deliverables heading.

The Report Properties module looks like the following:

A REPORT'S PARAMETERS (SELECTION CRITERIA)

Report parameters let a report restrict the records that it retrieves to only those records that meet specific criteria. When a report includes parameters, they appear in the Report Properties window shown previously.

For each parameter you can supply a corresponding value that will be used whenever the report is generated. You may answer each parameter in two ways:

- By manually filling in its value in the **Answer** field, i.e., enter the start/end values for a range.
- By selecting a date substitution variable, such as "today", "tomorrow", or "last Monday."
- Click on the green List button that appears next to the parameter you are working with and select the predefined variable you wish to use:

10.8.4 Report Delivery Methods

Reports can be emailed, FTP'd or linked to a webcast (dashboard) page. (See "Report Delivery Methods" later in this chapter for details.)

10.8.5 The Report's Subscribers

Reports can be delivered to any combination of individual subscribers and subscriber groups.

10.9 Triggered Reports

Unlike a scheduled report, a triggered report gets generated and delivered only if the event's triggering conditions are met. For example, you might want KnowledgeSync to generate an A/R Aging report for customers who are over their credit limit. This would require a triggered report, because only if there are any customers over their credit limit would you want the report to be generated.

An additional aspect of a triggered report is that the contents of the report(s) can be driven by the records that meet the event's criteria. Using the preceding example, an event's trigger would determine which customers are over their credit limit and then the event would take those results and feed them into the report to be generated.

Thus, you could have an event that generates individual A/R Aging reports only for those customers that are over their credit limit.

10.9.1 Triggering a Report on True/False Results

Many organizations need to generate a report if a situation proves true or false. Consider the following two examples:

Run a report if there were any sales of over $10,000 last week
Run a report if there were no sales of over $5,000 last week.

To configure either of these conditions, you would design a trigger that uses the "count" function to count the number of records that meet the query's criteria. In the first scenario, the report would trigger if the count of sales is greater than zero; in the second scenario, the report would trigger if the count of sales was equal to zero.

10.9.2 Linking a Report to a Triggered Event

To have an event generate reports that are based on the records that meet the event's criteria, you must have some way to take the results from an event's trigger and feed those results into the event's report. This is accomplished through a report's parameters. Consider the previous example – you have customers who are on hold and you want to generate an A/R Aging report for each on-hold customer. You would do the following:

Create a KnowledgeSync trigger (such as a query) that retrieves a list of on-hold customers.
Create an A/R Aging report so that is has a parameter on the customer ID.
Create a KnowledgeSync event that uses the trigger and report mentioned previously.
When you add the report to the event, the parameter (e.g., "please select customer ID") will appear. In the "answer" field to this parameter, you'll select the "customer ID" field from the event's trigger.

When the event runs, it will:

1. Use its trigger to generate a list of on-hold customers – e.g., Client A, Client B, and Client C.
2. Take the first matching record – Client A – and pass that client's ID into the A/R Aging report.
3. The Aging report for Client A will be generated and delivered.
4. Repeat steps #2 and #3 for clients 'b' and 'c'.

10.9.3 Delivering a Triggered Report to the Record "Owner"

Consider the previous scenario – generating a separate report for each client who is on hold. In addition to generating a separate report per customer, you might want to deliver that report to a "related party" – such as to the customer who's on hold, or even to the salesperson who's responsible for that client's account. A triggered-event enables this kind of intelligent delivery.

In the design of the "On Hold Customers" trigger, as long as you include the field(s) that contain the delivery addresses of the "related parties", you'll be able to tell KnowledgeSync to deliver the report to them. Using the above example, let's say that you want to email the A/R Aging report to both the customer and their account manager.

All you would need to do is to go into the event's trigger and make sure that it retrieved:

The field that contains the email address of the customer
and;
The field that contains the email address of the customer's account manager

By including those two fields in the event's trigger, you'll be able to select them as "advanced subscribers" for the KnowledgeSync triggered report event.

10.9.4 Trigger Design for Triggered Reports

It's important to keep in mind that KnowledgeSync will (by default) generate one report per record that meets an event's criteria. So – build your trigger so that it retrieves only one record per report that the event will generate. Consider the following example:

Your staff gets assigned activities; sometimes those activities become overdue. You want KnowledgeSync to see if a staffer has overdue activities and (if so) generate and deliver a "Past Due Activity Report" to them.

Your trigger should not generate a list of past due activities, because if a staffer has 10 past due activities, KnowledgeSync will generate 10 reports for that person. Instead, configure your trigger to count the number of overdue activities per staffer and retrieve only those staffers whose "past due count" is greater than zero. That way, regardless of the number of past due activities, each staffer with past due activities will appear only once in the trigger's result set – and only one report will be generated and delivered to each of them.

10.9.5 Using Triggered Reports to Generate & Deliver Forms & Documents

A very common use of triggered reports is in events that generate and deliver forms and documents such as invoices, statements, order confirmations, shipment confirmations, purchase orders, and the like.

In these scenarios, your report is actually a form (such as the ones mentioned previously), and it has a parameter that prompts for the "record ID" – such as a sales order number, customer number, or purchase order number. The event's trigger retrieves a list of matching records (new orders, new shipments, new POs, etc). When you link the report to the event you will pass the record ID (e.g., the sales order number from the trigger) into the report as a parameter.

10.9.6 Begin & End Repeat in Triggered Reports

In most cases, an event-triggered report will not make use of {BEGIN*REPEAT} and {END*REPEAT} in its alert message. In such cases, the event will generate one report for each record that meets the event's criteria.

However, if an event uses {BEGIN*REPEAT} and {END*REPEAT} to group multiple records within separate messages, KnowledgeSync will generate one report per delivered message. (Thus, an event that puts all matching records in a single

alert message would, if it had an associated report, generate and deliver only one report, regardless of the number of records that meet the event's criteria.

10.9.7 Trigger Design for Triggered Reports

Keep in mind the following 3 concepts when designing a trigger:

Use filters to control your result set – remember that KnowledgeSync will generate one report for each matching record, so it may be more effective to use the "count" aggregate function so that you do not generate the same report multiple times.

Be sure to include the columns/fields whose values need to be passed into the event's associated reports as parameter values.

Also include the columns/fields that contain the addresses of potential report recipients – such as the email addresses of the related customer, prospect, vendor, salesperson, or manager.

10.9.8 Report Selection

To link a triggered report to an event, follow these steps:

Open the event you wish to associate the report to.
Go to the event's "Deliverables", click on the blue plus sign, and add the desired report to the event.
Review the report's "properties" and select the type of output format from the following:

- o   Image
- o   ASCII text
- o   HTML
- o   MHTML
- o   Rich Text
- o   Microsoft Excel (XLSX)
- o   Microsoft Excel legacy format (XLS)
- o   Comma Delimited
- o   Tab Delimited
- o   PDF
- o   Microsoft Word
- o   Crystal
- o   XML

(The output choice of "Crystal" is usable only by users who have the Crystal Reports "viewer" installed on their client workstation.)

Fill in the answers of any report parameters that appear.

For each parameter you can choose to manually fill in the answer or click on the List button to choose either a date substitution variable or a field value from the event's query.

You may add additional reports to a single event.

10.9.9 The Triggered Report's Schedule

Like any standard KnowledgeSync event, a triggered report event runs on a periodic schedule (e.g., hourly or daily) to see if the conditions specified in the event's trigger(s) have been met. If they are met, the event is triggered and the corresponding reports are generated and delivered.

Click on the description beneath the "Schedule" heading and select the schedule that controls how often this event is submitted.

10.10 Report Delivery Methods

Reports can be emailed, FTP'd or linked to a webcast (dashboard) page. In some instances, it may be most efficient to use email (or SMS) to <u>notify</u> recipients about the availability of reports while using FTP or webcast to <u>deliver</u> those reports.

10.10.1   Email

At the bottom of the Email Deliverable window arethree checkboxes that allow you to specify whether to attach the selected reports to the outgoing email message.

If you "include reports as email attachments" or "include files as email attachments", KnowledgeSync waits until the reports are successfully completed before sending the mail message. If a report fails to complete successfully, the email message will not be sent.

If you choose to <u>not</u> attach the reports, you can specify whether the email message should wait until all the associated reports are successfully generated. (Place a checkmark in "Email notifications should wait until reports are completed.")

10.10.1.1 Embedding the Report Within the Email

If you are sending mail in HTML format, an event may be configured to send the report's contents within the <u>body</u> of the email message (as opposed to sending the report as an attachment).

This function is available <u>only</u> if you have KnowledgeSync configured to send out mail via SMTP or any other INTERNET-based mail system.

To configure this, follow these steps:

Choose the report whose content you wish to appear within the body of the email message. (You can specify only a single report.)
Choose a report output type format of "HTML".
Go to the "Email" tab and leave the "Message Text" field blank (empty).

When the event is triggered, KnowledgeSync will generate the corresponding report in HTML and will embed this HTML within the body of the email message that is sent to the event's subscribers.

If a subscriber of such an event is using an email system that does not support HTML, the report will appear as an attachment. Note also that such an event cannot have any other attachments (i.e., no other reports and no attached files).

10.10.2   FTP/SFTP (Copy)

To deliver a report via this method, go to the event details window and select the option to "include reports on copy operations". Note that the copy/ftp destination is determined within each subscriber's profile in KnowledgeSync. Specify this location in UNC format and do not reference mapped drives.

10.10.3   Webcast (Dashboard)

When a report is configured for delivery, KnowledgeSync enables you to specify whether the report will appear as a URL link in a webcast (dashboard) alert message. (Please see the chapter on "Webcasting / Dashboard Alerts" for more details.)

At the bottom of the Webcast Deliverable window are two checkboxes that allow you to specify whether to link the selected reports to the generated webcast page.

If you "include reports as URL links", KnowledgeSync will wait until the reports are successfully completed before generating the webcast page. If a report fails to complete successfully, the webcast will not be generated.

If you choose to <u>not</u> link the reports, you can specify whether the generation of the webcast page should wait until all the associated reports are successfully generated. (Place a checkmark in "Webcast notifications should wait until reports are completed.")

10.10.4 Report Subscribers

Reports can be delivered to any combination of standard and advanced subscribers; both individuals and groups. Because a triggered report has associated triggers, all the trigger data fields may be used to contain recipient alert delivery addresses.

10.11 How Reports Appear in the Monitor

Scheduled reports start out in the tab called "Report Generation" (they go from "pending" to "generated") and then move to the tab that represents how the reports are to be delivered to their intended recipients (e.g., "Email Delivery").

Triggered reports begin in "Application Events", from where you can track the event's status as "checked" and (if appropriate), "triggered." Only if an event is triggered will its associated report(s) be generated.

Once an event with associated reports has been triggered, it will proceed to the "Report Generation" tab. From this branch, you may track the progress of the event's associated reports as they go from a "pending" to a "generated" state.

(If a report fails to generate successfully, it will appear under "Errors" in the "Report Generation" tab.)

Once a report generates successfully, you may click on the corresponding "Delivery" tabs within the Monitor and check to see that the message (and report) was successfully delivered.

Within the Application Events → Triggered tab, you may also click on the "Details" (blue plus sign) next to an event to see whether the notifications included the corresponding reports.

11 Generating & Delivering Charts

KnowledgeSync includes an embedded chart & graph design module that adds visual detail to an alert. Like Reports, charts can be scheduled, generated, and delivered (such as a "Daily Open Tickets" chart) or they can be associated with events (such as an event that checks to see if any support reps have more than 10 tickets assigned to them and – if so – generates and deliver a "Rep Ticket Load" chart.

11.1 Charts, Events, & Queries

It is very important to understand that a chart uses a query to generate its content. Do not confuse a chart's query with an event's query – charts and events rarely (if ever) use the same query, and it is <u>not</u> recommended that they do so.

As such, it is suggested that any queries that you design for use with charts be given a unique naming convention, such as starting the query name with "Chart; . . . " as in "Chart; Calls Per Rep".

11.2 How Charts Are Generated

Charts have their content driven off of the results of a query; for example, a chart that shows "current month total sales by salesrep" would use a query that totals sales by salesrep and then has a filter to retrieve only those orders that have been placed during the current month.

Likewise, a chart that shows how many support calls are currently assigned to each support rep would use a query to count the number of open tickets per rep and would use a filter to retrieve only those tickets that were currently open.

Charts usually use aggregate queries (queries that either sum a numeric field or count the number of records meeting certain criteria), but charts can use non-aggregate queries as well (such as a query that shows all currently backordered line items and their backorder total).

11.3 Chart Query Design: Two Requirements

Regardless of whether a chart is using an aggregate query or a not, a chart's query should always retrieve just two columns – and in a specific order:

1. Column #1: The chart's horizontal (or 'x') axis; the "item being counted", such as "customers" (whose sales are being totaled) or "support tickets" (which are being counted per support rep).

2. Column #2: The chart's vertical (or 'y') axis; the numeric representation. This is the query column performing the aggregate function, such as "summing" sales totals, or "counting" the number of support tickets.

11.4 How Charts & Graphs Are Delivered

Charts and graphs can be delivered in the following manners:

Attached to an email (you can attach multiple charts to a single email)
Embedded within the body of an email (only one chart may be embedded within a single email message)
Posted to a webcast page
FTP/SFTP to a destination folder

11.5 Chart Design: The Role of "Parameters"

A chart can use "parameters" in the same way a report does. Consider, for example, an event that generates a sales report -- one report per salesrep, detailing their monthly sales. The report would have a parameter on the salesrep ID.

An event that generates this report would use a "salesrep listing" query to retrieve a list of salesreps. Each of those salesreps would be passed one-at-a-time into the report's parameter and so generate one report per salesrep.

Charts work in a similar manner.

You'd begin by creating the chart; it would use its own query and would have a parameter (filter) on the salesrep ID. The filter's "compare value" would be the question mark ('?') .

An event that generates this chart would use the same "salesrep listing" query to retrieve a list of salesreps. Each of those salesreps would be passed one-at-a-time into the chart's parameter and so generate one chart per salesrep.

A chart's parameters (filters and/or subfilter) appear when you select the chart for an event. Because an event has its own query, you can select data fields from the event's query to be passed into the chart's parameters.

And so, when an event with a parameter-driven chart runs, it will:

1) Use the event's query to retrieve the first matching record.
2) Pass a value from that record to the parameter of the chart associated with the event.
3) Generate the first chart and deliver it.

4) Repeat steps 1 – 3 for each ensuing record retrieved by the event's query.

11.6 Charts: Scheduled & Triggered

Charts can be underlined scheduled (such as a "daily sales chart"), or charts can be triggered (such as a "tickets per rep" chart if any rep has more than 10 tickets assigned to them).

A scheduled chart event has no query associated with it.

A triggered chart event is an event that uses a query to determine if certain conditions are met; if so, the associated chart is generated. Triggered charts will be generated once for each record that meets the event query's criteria.

11.7 A Chart's Query

Let's consider the following chart:

This month's sales per salesrep

This chart will total the current month's sales per salesrep; this chart (as with most) will make use of an aggregate query.

The query for this chart will retrieve sales numbers and salesperson details. The chart will have two columns (the "salesrep's name" and the "sales amount". To "total" the amount for each salesrep, the "amount" field will have its column type changed from "Normal" to "Summarize".

The query will have a filter to retrieve only those sales that were placed during the current month. This filter could be parameter-driven (question marks in the "Compare Value" field) and other filters could be added.

> *Based on the chart "type" (column or pie), you may want the data to appear in the chart in a specific order, such as showing the greatest amount at either the top or bottom of the chart. To accomplish this, you will need to specify the "Sorting" order for the chart's query.*

11.7.1 Create the Query

The first step in creating a chart is to create the query that powers the chart. As stated previously, a chart's query should have only two (2) columns:

1. Column #1: The chart's horizontal (or 'x') axis; the "item being counted", such as "customers" (whose sales are being totaled) or "support tickets" (which are being counted per support rep).

2. Column #2: The chart's vertical (or 'y') axis; the numeric representation. This is the query column performing the aggregate function, such as "summing" sales totals, or "counting" the number of support tickets.

For example:
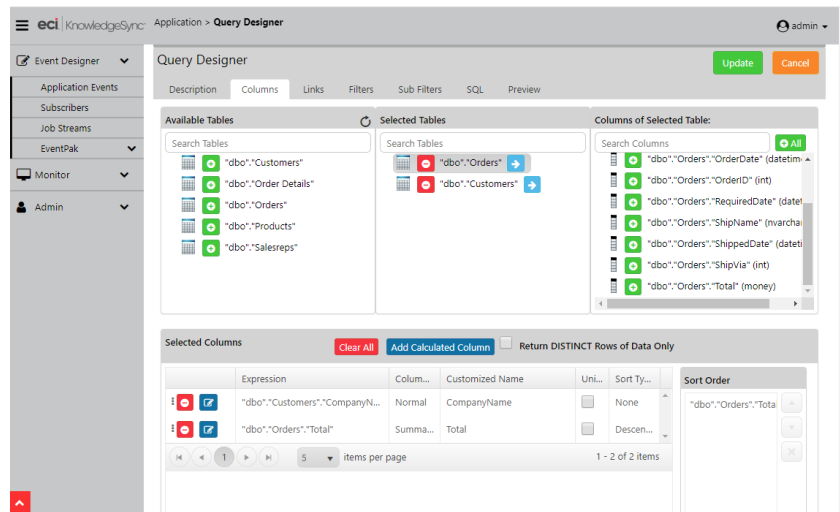
CHART QUERY DESIGN

And the typical results of a chart's query will look like the following:
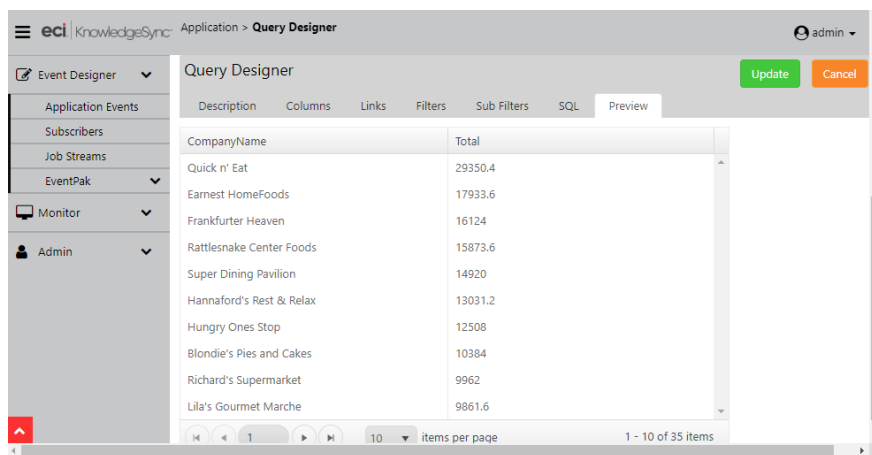
CHART QUERY OUTPUT

Once you have designed and reviewed a chart's query, you can proceed with the configuration of the chart itself.

11.8 Chart Design: Create the Chart

Open the Event Designer, expand the application you wish to create a chart for, single-click on the "Charts" sub-branch and then click on the "New Chart" button. The following window will appear:

CHART DESIGN WINDOW

Make your selections based on the options detailed in the following section.

11.8.1 Chart Design: Formatting Options
Select from the following options:

Chart Name: Give the chart a name that describes its contents, such as "Sales; Current Month Per Rep".
Chart Type: Choose from two formats; "pie" or "column" (bar).
Chart Style: The "style" options for a chart will vary based on the "type" of chart selected. For example, bar charts can be configured so that its bars appear "flat" or in "3D"; pie charts have additional style options.
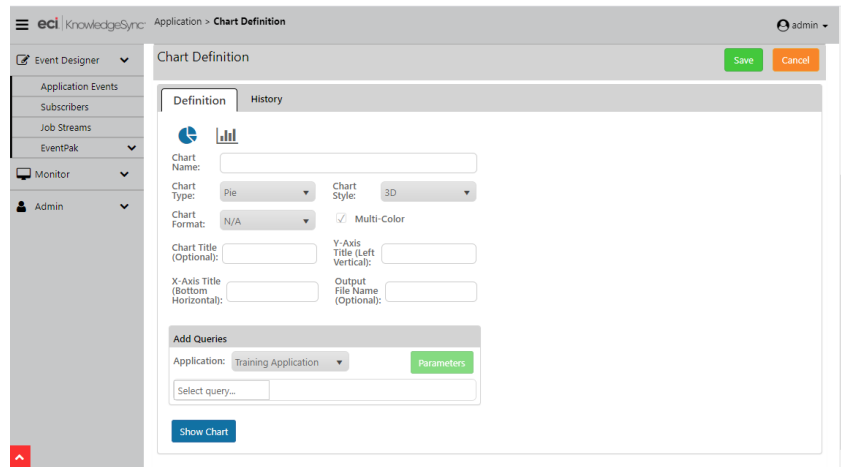**Chart Format**: (Not currently enabled)
Multi-Color: Check ON for multi-color display.
Chart Title: This is the title that will appear across the top of the chart. You may leave this field blank.
Y-Axis Title: This is the title that will appear along the left (vertical) side of the chart. Typically, this would have a title such as "Total Sales This Month". You may leave this field blank.
X-Axis Title: This is the title that will appear across the bottom of the chart. Typically, this would have a title such as "Salesperson Name". You may leave this field blank.
Output Filename: This is the name that the chart file (once generated) will be saved to. Note that all generated charts will (by default) be saved to the KnowledgeSync "attachments" folder; once in that folder, KnowledgeSync can move (or "copy") the chart file to any destination location.

When choosing an output filename, you may hard-code the name or you may use any of the variable values (date variables & query variables). Do not specify an output filename extension, such as "jpeg" or "bmp", as KnowledgeSync will automatically append that extension based upon the selected output format of the chart.

11.8.2 Query Selection
Under the "Add Queries" heading, select the KnowledgeSync application that contains the query that will be used to drive the chart's contents. Once you select the application, you can proceed with selecting the specific query that will populate the chart's contents.

You have direct access to edit the selected query (click on the blue pencil icon) as well as remove a selected query from a chart (click on the red minus sign).

11.8.3 Query Parameters
If the chart's query includes "parameters" (filters or sub-filters where the "compare value" is a question mark), the green "Parameters" button will be enabled; if the query has no parameters, this button will be greyed-out.

Consider a chart called "Sales Per State by Salesrep". The query that powers this chart would, for a selected state, show the sales per salesrep. This query would have a filter on "state" and the answer (compare value) for the selected "state" can be specified in one of 4 places:

You can hard-code the answer in the chart's query. When you configure the chart's query and create the filter on the "state" field, you could type in answer, such as "WA" for the state of Washington.
You can hard-code the answer in the chart. When you select a chart's query and that query includes a filter where the compare value is a question mark, you will be prompted to supply an answer to that parameter. In the above example, you would click on the chart's "Parameters" button and type in "WA" as the selected state.
You can hard-code the answer when the chart is selected for use in an event. By leaving the '?' as the answer to a parameter in the chart designer, you will be prompted to specify the parameter's answer when you select the chart as part of an event. In this case, you could select "WA" as the chart's parameter answer for when the chart's event is run.
You can have an event's query pass its values into a chart's parameters. Let's say that you want to generate this chart only for states that have recorded more than $10,000 in sales. Your event would use a query to retrieve a list of those states – and then that event would pass those "state values" to your chart, one-at-a-time.

In such a scenario, instead of hard-coding the state value, you would select the chart for this event, click in the "Answer" field to the chart's "state" parameter, and select the query data field that contains the matching state names – this query data field typically looking like this: {state}. (This way, if there were 5 states that met this event's criteria, the event would auto-generate 5 charts; one for each state.)

A chart's parameter entry window looks like the following:
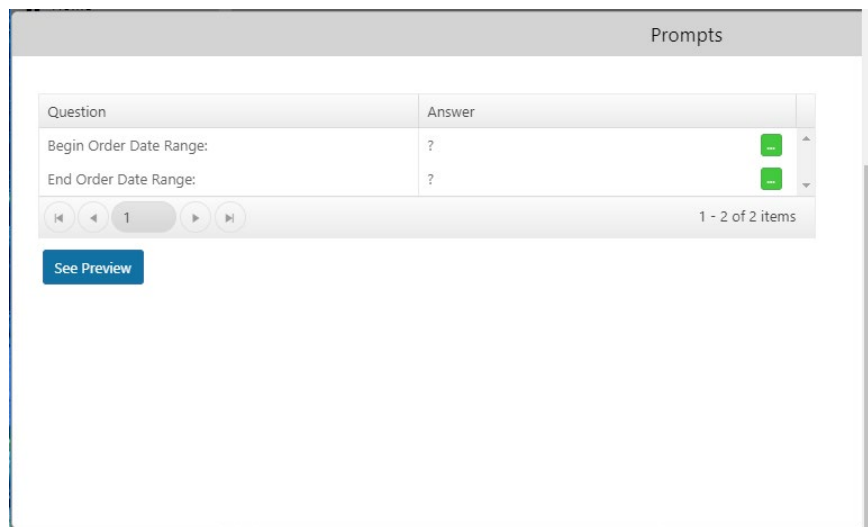
CHART PARAMETER ENTRY

Answer any query parameters and remember that you can leave parameter answers with the '?' symbol to indicate that you will supply a value when the chart is configured within an event.

11.9 A Completed Chart ("Show Chart")

When you are ready to test or preview your chart based on the configuration selections you have made, click on the button called Show Chart". A window like the following will appear (using your selected chart type, style, and content):
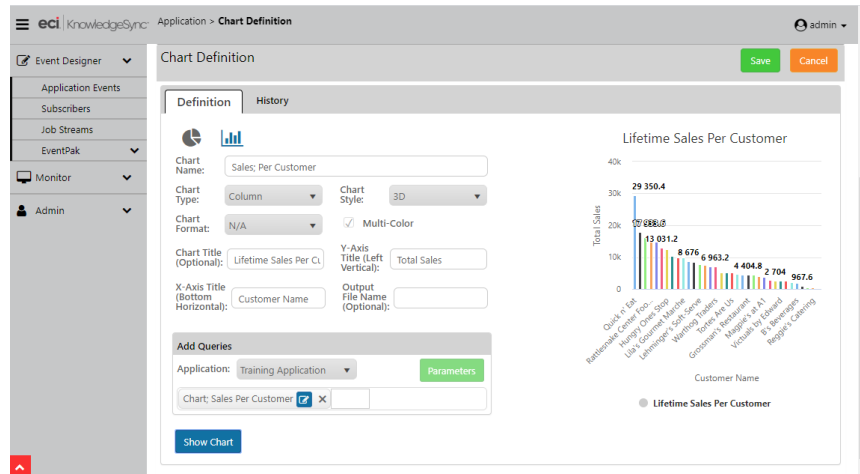
"SHOW CHART" OPTION

While viewing the chart, you may move your cursor to different elements of the chart to get additional details. Click on the "Save" or "Update" button to save your work.

11.10 Link a Chart to an Event

Once you have created and previewed your chart, you may add it to an event. Follow these steps:

Select the option to create a new event.
Select a trigger (e.g., a query, script, or web API) for the event (if the event is monitoring for specific conditions). If the event's purpose is to generate a chart on a recurring basis, do not select any triggers. If the query has any parameters (selection criteria), fill them in as needed.
Click on the "Plus" sign in the "Deliverables" section of an event and select "Chart" as alert content. Select the application in which the chart resides, select the specific chart to associate with the event, and click on the "Add Chart" button.
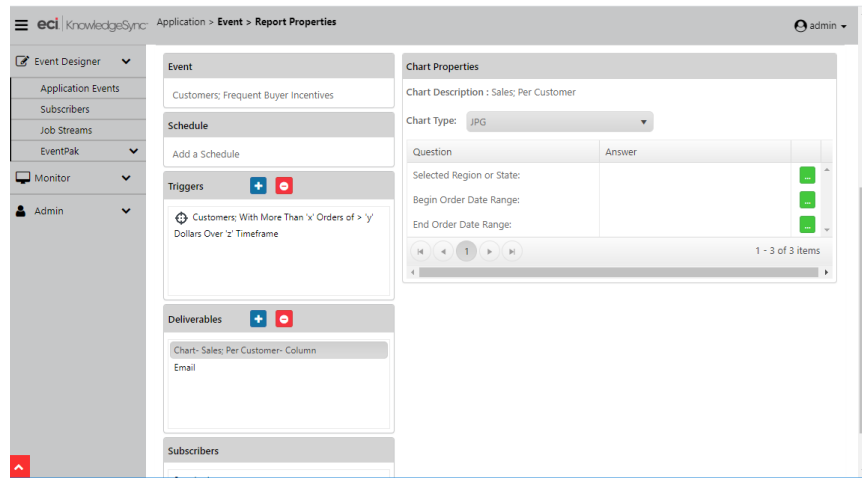
11.10.1   Select the Chart's Output Format

Select (single-click on) the event's chart deliverable and then (on the right-side of your window) choose from the following chart types (formats):

- PDF
- PNG
- JPG
- SVG

11.10.2   Supply Chart Parameter Answers

If the selected chart has any parameters, you'll see a window like the following:

AN EVENT'S CHART'S PARAMETERS

Parameters may be "answered" using any combination of hard-coded values, date substitution variables, and values from the event's triggers (e.g., query data fields).

11.10.3   Chart Delivery

Charts may be delivered to recipients as email attachments, embedded within an email (select output style "png"), via FTP/SFTP, and via web dashboard (webcasts) – in which the charts show up as URL links. Please note that all references to "reports" within an event's tabs apply equally to an event's charts.

Once you have selected how the chart is to be delivered, you may proceed with selecting the event's subscribers (alert recipients) and select the schedule that controls how often the event will be executed.

When a chart is generated, the chart's file is automatically generated in the KnowledgeSync "attachments" folder. Once there you may use a subsequent event (optionally in a job stream) to move the chart file to a destination folder of your choice.

11.11 Date-Sensitive Charts

One of the most powerful uses of charts is to display "performance over time" graphs. For example:

Sales this month versus last
Number of support calls logged daily this week
Scheduled phone calls per hour

For charts such as these, the corresponding queries are typically going to use a column that either "sums" a value (such as the amount of a sale) or a column that "counts" the number of records (such as the number of support calls) over a specific time range and – most importantly – "broken" by a specific date or time segment, such as yearly, monthly, daily, or hourly.

There is a very handy SQL command that lets you retrieve only the "year" or "month", or "day" (number; from 1 - 31) from a date field in a database. These functions look like this:

```
day(sysdba.ACCOUNT.CREATEDATE)
month(sysdba.ACCOUNT.CREATEDATE)
year(sysdba.ACCOUNT.CREATEDATE)
```

In addition to the above commands, the below SQL command enables you to retrieve the "day number of week"; from '1' (Sunday) to '7' (Saturday):

```
datepart(weekday,sysdba.ACCOUNT.CREATEDATE)
```

Whereas the following SQL command will show you the "name" of the day of the week:

```
datename(weekday,sysdba.ACCOUNT.CREATEDATE)
```
You can combine (concatenate) the preceding two commands together so that you can have both the day number and the day name in a single field in your query:

```
convert(char(1),datepart(weekday,sysdba.TICKET.CREATEDATE)) + '-'
+datename(weekday,sysdba.TICKET.CREATEDATE)
```
(The "convert" function is also used in the above command because the "combined" field has to be an alphanumeric field in order to contain both the day number and day name.)

Here is the SQL command for retrieving the "hour" from a date/time field:
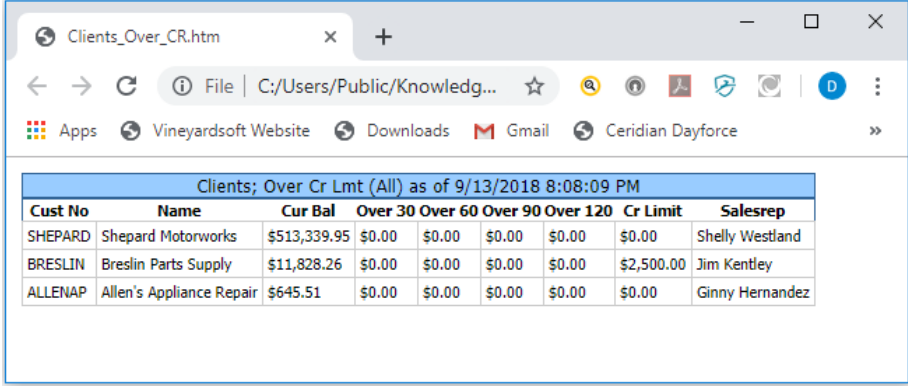
```
datepart(hour,sysdba.ACCOUNT.CREATEDATE)
```
The above command will show the hour in 24-hour (military) time.

All the preceding commands are for use when monitoring a Microsoft SQL Server database; if monitoring a database other than SQL Server, please refer to your database documentation on the use of date-related "scalar functions".

12 Dashboard Alerts (Webcasting)

Dashboard alerts are dynamically created and updated web (HTML) pages that contain triggered event details as shown below:

A free "Dashboard Setup Kit" is available from the Documentation page of the KnowledgeSync website.

12.1 How Dashboard Alerts Are Used

Dashboard alerts have three distinct uses:

Task-Specific Dashboards. This is a display of information that pertains to a specific business area – such as "overdue invoices" "low stock", "unassigned leads", or "open support calls".

Staff & Task-Specific Dashboards. This is a display of information that pertains to a specific user and their tasks – such as "my clients' overdue invoices", or "my activities for today".

User Alert Listing. This is a display of all alerts – for all business conditions – that apply to a specific user (e.g., "My Alerts")

12.2 What a Dashboard Consists Of

A dashboard consists of the following components:

- A template which controls the overall design of a dashboard window
- An event which controls the data within a dashboard
- Subscribers specify where dashboard output is generated.

The following steps will walk you through the creation of a single dashboard; one that displays a list of invoices that are currently overdue for payment.

12.3 Enable Dashboard Creation

To enable dashboards, follow these steps:

Expand the KnowledgeSync Monitor option and click on the "Activity" tab.
Make sure that the "Webcaster Server" shows a status of "ON". ("webcaster" = "dashboards").
If the webcaster server shows a status of "OFF", click next to the word "off" to enable this server.
Restart the KnowledgeSync Windows service.

12.4 Create Template

All dashboards have a general layout and design that is controlled by an HTML template. We suggest that you use the standard dashboard template from KnowledgeSync within the "Dashboard Set-Up Kit". The kit is available from the KnowledgeSync website's Support page, under Documentation:

https://www.vineyardsoft.com/downloads-documentation.php

In this kit is the default "dashboard template" which you may, optionally, edit (to include your company name, logo, etc). Note that you must use an HTML editor to make such changes. (Do not edit the template in MS Word, as it will lose its formatting.) You may edit a template using "Notepad" if you have no HTML editor available.

The only "variable" field in this template is called {EVENT_MESSAGE}. This field contains the contents of the "Webcast" deliverable of the event which generates a dashboard.

12.4.1 Storing the Template

We suggest storing your webcast template(s) in a folder called "**Webcaster**" – you will need to <u>create</u> this folder – and we suggest you do so directly under the KnowledgeSync installation folder, as in the following example:
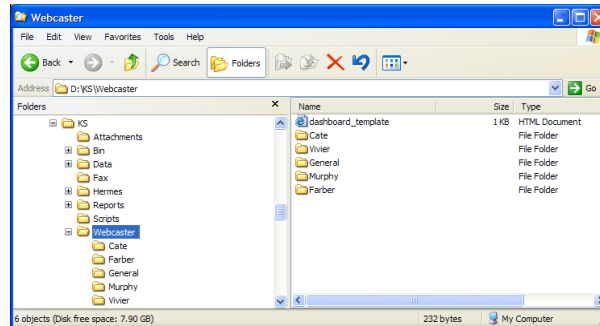
DASHBOARD TEMPLATES LOCATION

12.4.2 Creating Multiple Templates

You may wish to create multiple templates for the following reasons:

To have different dashboard layouts for different recipients. E.g., you might have one template that is used for your internal dashboards, but a different template for dashboards that are going to be seen by customers or suppliers.
To have dashboards that reside in different locations for security reasons. Each template has its own "publishing location" and so internal dashboards might be stored inside a firewall, whereas customer dashboards might be stored outside it.

To create multiple templates, make a copy of the original template, make any necessary changes to the new one, and then save the new template with its own unique name.

12.5 Create Publishing Location

You must specify where (system folders) dashboard output is to be generated and stored. This is referred to as a dashboard's "publishing location".

Dashboard output may be stored in one folder or in multiple folders. For example, if you wish to create dashboards for users "Smith", "Jones", and "Parker", you might wish to store these dashboards in corresponding folders called "Smith", "Jones", and "Parker".

Plus, if you were creating dashboards for various departments within your organization, you might create folders such "Sales", "Support", and "Shipping".

KnowledgeSync suggests that dashboards be created <u>in sub-folders beneath the KnowledgeSync "webcaster" folder</u>; you may specify other locations if desired.

12.5.1 Add Publishing Location Folder(s)

The steps below show the creation of three (3) publishing locations beneath the KnowledgeSync "webcaster" folder. To create your own publishing locations, follow these steps:

Open Windows Explorer and navigate to the folder where you installed KnowledgeSync.

Navigate to the "Webcaster" folder.

Create a new folder (beneath the Webcaster folder) and give it a name according to the person or group who will be receiving dashboard alerts.
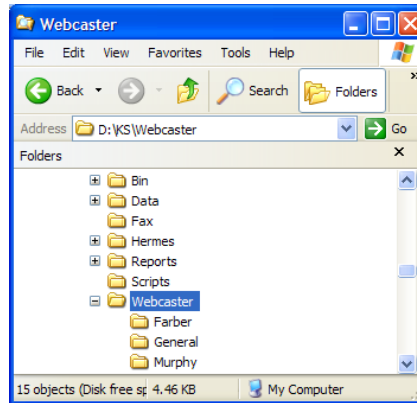
3 PUBLISHING LOCATIONS
("FARBER", "GENERAL", & "MURPHY")

12.5.2 Define "Default" Template & Publishing Location

Although you may define multiple templates and publishing locations, <u>the dashboard alerts module requires that you create a "default" template and publishing location record</u>. This becomes the template that KnowledgeSync uses for every event if no other template is specified.

Follow these steps to define the default template and publishing location:

Expand the Administrator branch and select the "Webcast Locations" branch. Click on the "Add" button to define a new location or on the blue Edit button to select an existing location:
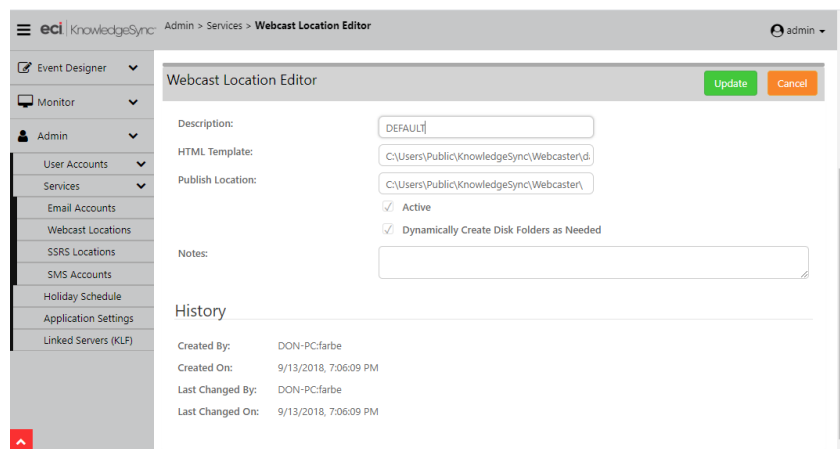
WEBCAST TEMPLATE & PUBLISHING LOCATION DEFINITION

Description: If you are adding your first webcast location, give this location a name of "**DEFAULT**" (all uppercase). You <u>MUST</u> have one template with a description of **DEFAULT**.

HTML Template: Specify the name and location of the default template that will be used. Do not reference mapped drives; use UNC specifications instead.

Publish Location: Enter the name of the folder <u>beneath which</u> you have defined the sub-folders in which webcast output will be generated.

For example, if you created publishing locations (folders) beneath the "Webcaster" folder called "General", "Murphy", and "Farber", your selection for the publish location would be:

> `D:\Program Files\KnowledgeSync\Webcaster\`

> (Do not use mapped drives and make sure you place a backwards slash at the end of the publishing location. This tells KnowledgeSync to put the dashboard output into the appropriate sub-folders <u>beneath</u> this location.)

Active: This makes the webcast publishing location available for use in KnowledgeSync events.

Dynamically create disk folders: This option lets you specify whether you want KnowledgeSync to <u>create</u> folders in which dashboards are created <u>if those folders do not already exist</u>. Note that the KnowledgeSync Windows Service must have the required privileges to create folders beneath the publishing location you specified above.

Note the following about publishing locations:

The computer running the KnowledgeSync service must have write capability to the publishing location.
You may specify an FTP/SFTP address as the publishing location using the format:

> `ftp://username:password@ftp.location.net/directory_name`

You may use a WML file as a template to support WAP access for handheld devices.

12.6 Dashboard Subscribers

Dashboard subscribers are most often controlled via the event's trigger; no other subscriber set-up is required. See the section of this chapter titled "Webcast Trigger Design" for details.

Only when a dashboard is used as a "User Alert Listing" are traditional subscribers used. Please see the section of this chapter titled "User Alert Listings" for details.

12.7 Webcast Trigger Design

Before you design a trigger for use with a dashboard, you need to decide the following:

Will my event create a single dashboard (e.g., "Customers on Hold") or multiple dashboards
(e.g., "Tim's Customers on Hold". "Dave's Customers on Hold", etc.)?

Your answer to the above question will determine the design of the trigger that is used with your dashboard event.

12.7.1 One Event – One Dashboard

If an event is going to generate a single dashboard, you should hard-code the location and name of that dashboard within the trigger itself.

Take an example of a "Customers on Hold" dashboard that displays a list <u>of all customers on credit hold</u>. You might decide to store this dashboard file in a folder called "General" and you wish to call that output file "customers_onhold". Here's how you would design the corresponding trigger/query:

Create a trigger/query that retrieves on hold customers.
Create a trigger/query column whose contents are:

> `'(name_of_publishing_folder)'+'\'+'(name_of_output_file).htm'`

> As in:

> `'General'+'\'+'customers_onhold.htm'`

The above column name tells KnowledgeSync to create the dashboard file in the "General" publishing folder, and to give the dashboard output file a name of "customers_onhold".

Give the above column a customized name of "dashboard_address".
Preview the trigger/query to make sure that the "dashboard address" field shows up as:

```
General\customers_onhold.htm
```

Save your trigger/query.
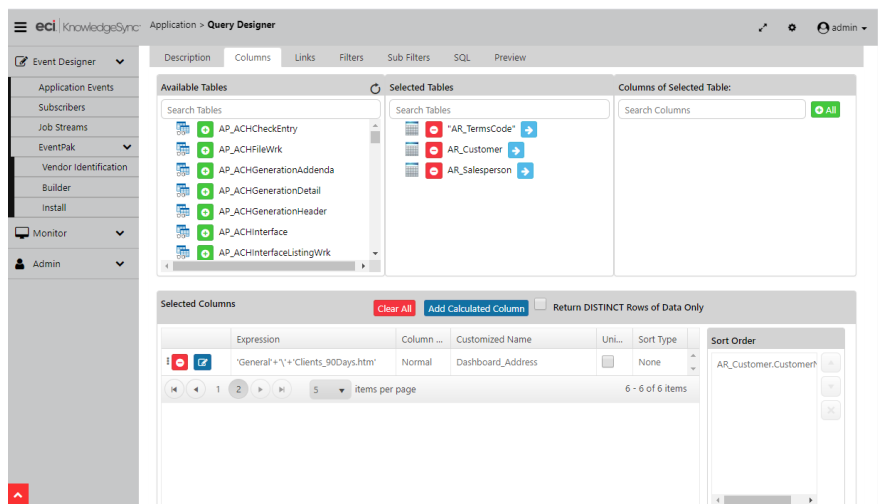
WEBCAST DASHBOARD QUERY WITH HARD-CODED PUBLISHING LOCATION

12.7.2 One Event – Multiple Dashboards

If an event is going to generate multiple dashboards (e.g., user-specific), you will need to embed <u>variables</u> within the trigger/query dashboard address column.

Using the example of the "Customers On Hold" dashboard, let's say you want to create one dashboard for each salesperson who has on-hold customers and you have salesreps named Farber, Smith, and Jones. You've already created a publishing location folder for each of them.

Now consider the trigger; each customer is associated with a salesrep (the three mentioned above) and each salesrep's last name corresponds to a publishing location. So – all you need to do is to tell KnowledgeSync to create a separate dashboard for salesperson who has on-hold customers.

You would create a calculated column, and its format would again be as follows:

```
'(name_of_publishing_folder)'+'\'+'(name_of_output_file).htm'
```

But instead of hard-coding the name of the publishing folder (as previously detailed), you would use a database field to control the name of the folder in which the dashboards will be created, as in:

```
"AR_Salesperson"."lastname"+'\'+'customers_onhold.htm'
```

"AR_Salesperson"."lastname**"** is the field that contains the last name of the salesrep to whom a customer is assigned – and that last name also corresponds to the name of the folder in which you want each dashboard to be created. KnowledgeSync will automatically create the appropriate dashboards in the appropriate locations based on the records retrieved by a trigger.

(And so, if 3 salesreps had on-hold customers, KnowledgeSync would create three dashboard output files, one in each of the 3 corresponding publishing location folders, and each file would be called "customers_onhold.htm".)

Triggers that create multiple dashboards <u>must</u> include a "Sorting" field that groups the matching records by dashboard recipient. Thus, in the above example, the query also includes a sorting field on the salesrep's last name.
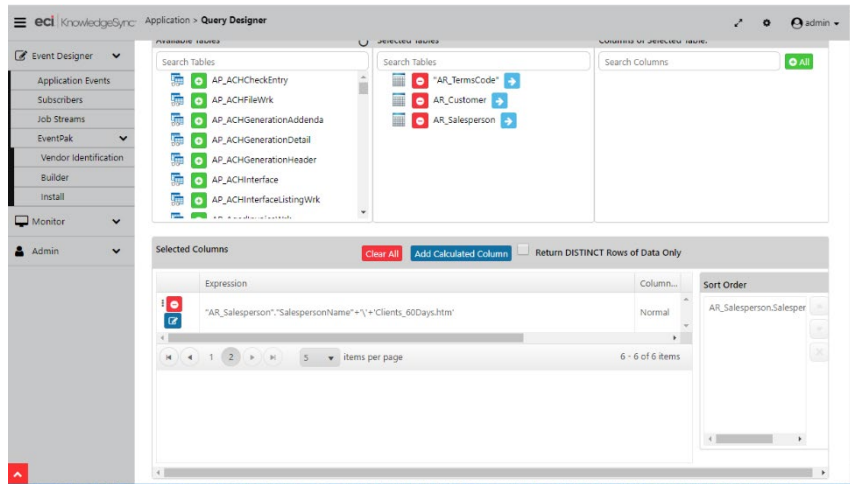
QUERY DESIGN FOR MULTIPLE DASHBOARDS (NOTE SORTING)

12.7.3 Dynamically Naming the Dashboard Output File

In addition to dynamically determining the publishing location of dashboard output files, KnowledgeSync also lets you dynamically name the dashboard output file itself.

In the example from the preceding section, 3 salesreps named Farber, Smith, and Jones would each have a dashboard file called "overdue_activities" in their corresponding folder. But what if you wished to precede the name of each dashboard output file with the name of the salesrep?

Easy – just modify the dashboard address column as follows:

```
(database_field)+'\'+(database_field)+'(name_of_output_file).htm'
```

As in:

```
"AR_Salesperson"."lastname"+'\'+"AR_Salesperson"."lastname"+'_customers_onhold.htm'
```

Now the salesrep named Farber will have a file called "farber_customers_onhold" in his folder, and the other salesreps will have correspondingly-named dashboards as well.

12.7.4 Choosing a Specific Dashboard Template

(You may skip this section if you are using a single dashboard template.)

Earlier in this chapter it was mentioned that you can create multiple dashboard templates. If you have multiple templates, you can control which template the trigger's resulting dashboard will use by modifying the dashboard address column as follows:

```
'(template_location_name):'+'(publishing_folder)'+'\'+'(name_of_output_file).htm'
```

As in:

```
dash_ext:General\overdue_activities.htm
```

This syntax will create a dashboard called "overdue_activities" in the "General" folder using a webcast template called "dash_ext".

12.8 Other Trigger Design Considerations

Consider the elements of data you wish to display within a dashboard. For example:

Query Designer					UPDATE CANCEL

| Description | Columns | Links | Filters | Sub Filters | SQL | Preview |

Vendor_Name	Item_Description	ItemNumber	MinimumOrderQty	QtyOnBackOrder	QtyOnHand
	AM/FM/STEREO	AM/FM STEREO	0	0	0
	AM/FM/STEREO/CASSETTE/CD PLAYR	STEREO/CD PCKGE	0	0	0
Container Corporation Of Usa	ART SPECIALTY EMP CANDLE LAMP	ARS-9401	10	0	-1
Container Corporation Of Usa	ART SPECIALTY GINGER JAR LAMP	ARS-9201	10	0	-17
Container Corporation Of Usa	ART SPECIALTY WALNUT CNDL LAMP	ARS-9101	60	0	21
Container Corporation Of Usa	ART SPECIALTY WALNUT CNDL LAMP	ARS-9101	10	0	-1
	BOOK BIN W/FLUORESCENT LIGHT	D1000-BBW/LIGHT	0	0	0
	BOOK BIN W/FLUORESCENT LIGHT	D1000-BBW/LIGHT	0	0	-1
Container Corporation Of Usa	CASTERS (SET OF 4, 2 W/LOCK)	VOG-CM-CASTERS	20	0	19
Container Corporation Of Usa	CASTERS (SET OF 4, 2 W/LOCK)	VOG-CM-CASTERS	200	0	101

DASHBOARD QUERY WITH MULTIPLE DATA ELEMENTS

Dashboards display their data in this kind of columnar format, and the number of data elements you can include in such a table is limited only by the size of the fields and the display capabilities of your monitor. (You can also adjust the font size of the data in the dashboard template.)

Regardless of how many data elements you wish to display, make sure your trigger/query has selected each of those data elements as "Columns". Only those columns that are selected for a trigger/query can be later put into a dashboard.

Dashboards can also contain calculated data such as:

Total sales per customer
Average sale amount over the last month
Count of orders per salesperson

12.8.1 A Dashboard That Uses an Aggregate Trigger

If you are creating a dashboard trigger/query that uses an aggregate function – such as the "summarize" or "count" functions, the column that contains the dashboard address <u>must</u> include a database field name (due to SQL syntax restrictions).

The best way to handle this is to choose a one-character field from your database (such as a "status" field) and use the "substring" command to retrieve a value from that field that will always be blank. For example:

```
SUBSTRING(sysdba.CLIENT.STATUS,20,1)+'General'+'\'+'Opp_Count_Value.htm'
```

In this example, no data is being retrieved from the "client status" field, but without a field reference, the dashboard address value (by itself) would not be accepted by the query's SQL syntax.

12.9 Configure Dashboard Event

To specify the content of a dashboard, create an event that uses the trigger you designed, and then select the "Webcast" deliverable in the Event Designer.

The **Webcast** deliverable is where you specify the contents of your dashboard. Dashboards are typically formatted to display their data in a table (columnar) manner; since this data will be shown in an HTML format, the webcast message must be defined using HTML syntax.

The webcast subject will typically be a description of the overall event – such as "Overdue Payments". You can also include data variables in your subject, such as "Overdue Payments for {rep_name} Clients"

A standard webcast message template is provided with the Dashboard Set-Up Kit from KnowledgeSync. We encourage you to use this template.

Click on "Preview Message" to create or customize the webcast content; the standard message content template provided by KnowledgeSync is:

```
<style type='text/css'>
     caption.mytable
     {
            background-color:99CCFF;
            color:black;
            border-style:solid;
            border-width:1px;
            border-color:336699;
            text-align:center;
     }

     table.mytable
            {
            font-family:Tahoma;
            border-collapse:collapse;
            font-size:10pt;
            background-color:white;
            width:100%;
            border-style:solid;
            border-color:336699;
            border-width:1px;
     }

     th.mytable
     {
            font-size:8pt;
            color:black;
            text-align:center;
            font-weight:bold;
     }

     tr.mytable
     {
     }

     td.mytable
     {
            font-size:8pt;
            background-color:white;
            color:black;
            border-style:solid;
            border-width:1px;
            border-color:cccccc;
            text-align:left;
            padding:3px;
     }
</style>

<table class='mytable'>
     <caption class="mytable">{EVENT_DESCRIPTION_COLOR} as of {EVENT_DATE}</caption>
```

```
<thead>
        <tr class='mytable'>
                <th class='mytable'>Heading 1</th>
                <th class='mytable'>Heading 2</th>
                <th class='mytable'>Heading 3</th>
                <th class='mytable'>Heading 4</th>
        </tr>
</thead>
<tbody>
        {BEGIN*REPEAT}
        <tr class='mytable'>
                <td class='mytable'>{Data Field 1}</td>
                <td class='mytable'>{Data Field 2}</td>
                <td class='mytable'>{Data Field 3}</td>
                <td class='mytable'>{Data Field 4}</td>
        </tr>
        {END*REPEAT}
</tbody>
</table>
```

The variable {EVENT_DESCRIPTION_COLOR} contains the name of the dashboard event and {EVENT_DATE} contains the date and time the event was last run.

The HTML table within the webcast message uses {BEGIN*REPEAT} and {END*REPEAT} so that all matching records will be displayed within the dashboard that's created.

12.9.1 Expiration Days

The field called "Expiration Days" controls the number of days that information will remain in the Dashboard. Typically, users do <u>not</u> want dashboard information to remain in the display unless the corresponding business conditions remain in existence.

An "Expiration Days" setting of **<u>minus one (-1)</u>** (the suggested setting) tells KnowledgeSync that each time the dashboard event runs, its contents will be refreshed with only those records that meet the event's criteria.

12.9.2 Linking Reports/Charts/Files to a Dashboard

Dashboards may be created to include URL links to reports, charts, and files. For example, you might have a dashboard that shows overdue payments and you might wish to include a report or chart that shows corresponding A/R Aging Report details.

To link a report, chart, or file to a dashboard event, follow these steps:

- Create the report, chart or file as needed.
  If a report or chart, define it within KnowledgeSync.
- Open the dashboard event, go to the **Deliverables**, and add the corresponding reports/charts/files. Select the output format and fill in any required parameters.
- Go to the **Webcast** deliverable window and place a checkmark in the option boxes titled:
  - Include reports as URL links
  - Include files as URL links
  - Webcast notifications should wait until reports are completed.
  (The report checkboxes apply to both reports and charts.)

The last step in linking a report, chart, or file to a dashboard is the process of specifying <u>where</u> – in the dashboard – the link to the report/chart/file (referred to as the report reference field) appears. The name of the webcast field that provides the link to associated reports/charts/files is {EVENT_REPORT_URL}, {EVENT_CHART_URL}, or {EVENT_FILE_URL}.

(Each report, chart, or file associated with an event appears as a separate URL link in the webcast.)

If the dashboard is generating just ONE report (and not one report per triggered record), place the KnowledgeSync variable called {EVENT_REPORT_URL} in the webcast's title (or caption):

```
<caption class="mytable">Overdue Invoices {EVENT_REPORT_URL}</caption>
```

If, however, you want the dashboard to generate one report <u>per triggered record</u>, you should place the {EVENT_REPORT_URL} value in the body of the webcast, as in:

```
<th class='mytable'>Order Conf Doc</th>
<td class='mytable'>{EVENT_REPORT_URL}</td>
```

This enables you to have reports and charts that are unique to each record in the webcast display. In the above example, the first line below is an HTML table's column heading, and the second line is for each table row, and will display the report/chart link that is related to the corresponding triggered record:

12.9.3 "Repeat-Enable" the Dashboard Event

Since you want to make sure that every time a dashboard runs, KnowledgeSync will show <u>all</u> matching records, make sure that the event is repeat-enabled by going to the event details window and checking ON the option to "repeat notifications".

12.9.4 Dashboard Subscribers

Except for user alert listings (detailed later in this chapter), dashboard events do <u>NOT</u> use Standard Subscribers. This is because the trigger/query that powers a dashboard event already contains the location and name of the dashboard output files; all you need to do is to specify the name of the trigger/query field that contains the dashboard location and name information.

(In the example from earlier in this chapter, the field where this was stored was called "dashboard_address".)

Thus, you would do the following to select the dashboard subscriber:

- Click on the Advanced Subscribers" option from within your dashboard event.
- Click in the last field in the lower grid ("Contains webcasting name") and a list button will appear.
- Click on this button and choose the field called "dashboard_address".
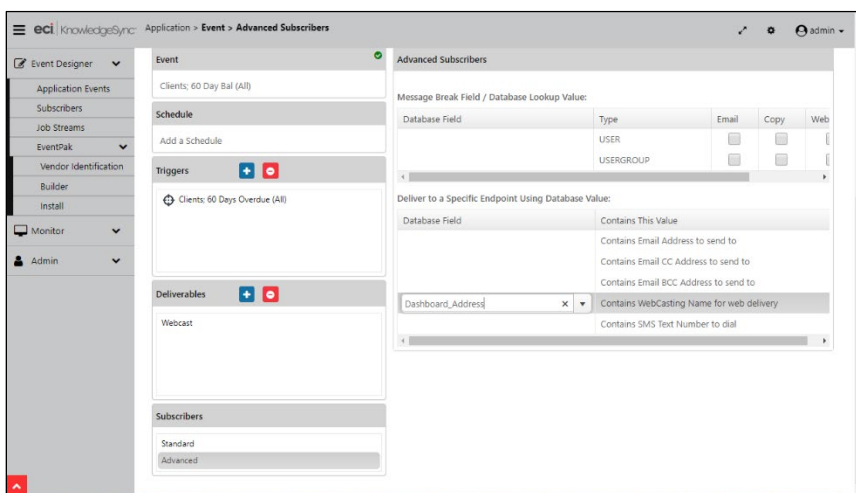
12.10 Schedule the Event

Like any other event, go to the "Schedule" tab to specify how often a dashboard event runs. Note that the frequency of running will determine how often this event updates its display with new or changed data.

12.11 Run a Dashboard Event

To run a Dashboard event once – right now – follow these steps:

Click on the desired dashboard event to highlight it.
Click on the button called "Run Now".

Once the event has been submitted, go to the Monitor module to track its status. Begin in the "Application Events" branch to see that the event has gone from "Checked" to "Triggered" (assuming that the event's conditions have been met.)

Once triggered, you can click on the blue plus sign ("Details") next to an event and then view its tracked items and deliverables. You can then go to the "Webcast Delivery" tab and look at the "Pending", "Sent Today", or "Errors" tabs to see the status of the generation of your dashboard.

12.12 A Portal for Reports & Charts

A second use for dashboards is as a "portal" where generated charts and/or reports can be published and then easily accessed by users.

Portals can be created per user, per department, or even company wide.

A sample template for the Reports Portal shown below is available from KnowledgeSync; please contact our technical support department if you would like to receive it.

When using webcasting for reports and charts, be sure to check ON the webcast window options for "including reports as URL Links." A simple reports & charts portal can look like the following:

SAMPLE REPORTS & CHARTS PORTAL

12.13 User Alert Listings

An additional use for dashboards is as a display of all of the alerts sent to a user.

In this configuration, a user would typically have a single dashboard window which would list all of the alerts from the events that the user wishes to be informed about.

Because such a window could contain a wide variety of event data, the design of such a window is typically very generic. A sample template (called "KS_Webcaster.html") is included in the "webcaster" folder beneath the KnowledgeSync installation folder. Typically, this same template would be used for any users who wish to have access to this display of their own triggered alerts.

Like all Dashboards, User Alert Listings are self-updating so newly-triggered events will automatically be added to the displays. User Alert Listings are also an excellent way to deliver Forms, Documents, and Reports to the users viewing these listings.

User Alert Listings typically look like the following:
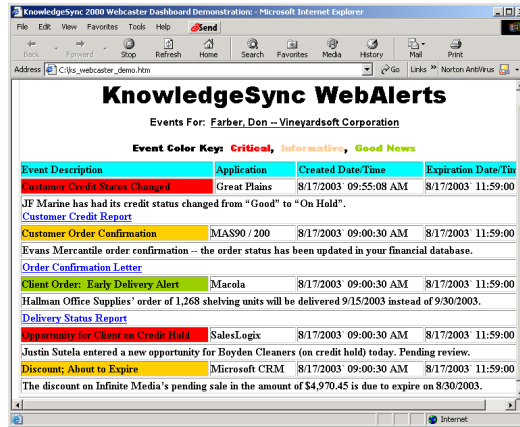
SAMPLE USER ALERT LISTING

12.14 Subscribers to User Alert Listings

Once you have created your webcast publishing locations, you can define the address for each subscriber who will receive a user alert listing dashboard.

Go to the KnowledgeSync "Subscribers" option, locate the subscriber whose address for their user alert listing you wish to define. Go to the "Webcast" tab for the selected subscriber.
In the "Name" field, type the name of the webcast folder in which the alert listing will be created. (Do not entire the entire path; just the publishing folder name.)
In the "Location" field, click on the drop-down arrow and select the dashboard template to be used when displaying this subscriber's alerts. You may define both a primary and secondary webcast delivery details for a subscriber.
If you will be using webcasting but not for user alert listings, it is suggested that you select the "DEFAULT" publishing location.

Webcast subscriber details looks like the following:

SUBSCRIBER WEBCAST ADDRESS DEFINITION

For any event whose webcast details you with to populate into a subscriber's User Alert Listing, go to the "Subscribers →
Standard" window of the corresponding event, select the subscribers whom you wish to generate the User Alert Listings
for, and place a checkmark in their "Webcast" delivery option.

12.15 How User Alert Listings are Populated

When an event that updates a User Alert Listings is triggered, KnowledgeSync automatically creates or updates the
corresponding dashboard file and publishes it to the corresponding location of each subscriber.

When additional events are triggered, KnowledgeSync automatically re-creates the user alert list dashboard file, including
both the existing alerts and the details of any newly triggered events. (Existing alerts will continue to be listed on a
subscriber's alert listing dashboard until the alert's "expiration date" has arrived.)

One of the most powerful aspects of user alert listings is that unlike email, a subscriber may keep a web browser open to
their specific alert listing location and receive real-time updates about newly-triggered events.

13 EventPaks

EventPaks are collections of KnowledgeSync events and may include triggers, events, reports, charts, and web service APIs. By building an EventPak you can "pack up" event-related data and send that data to another person or organization.

EventPaks have three primary uses:

To package custom events to share with other people or organizations
To move events from one system to another, or move events from a "testing" environment to "production".
To send events that you need help with to your business partner or KnowledgeSync

13.1 The Contents of an EventPak

Eventpaks contain event data that is associated with one or more applications within KnowledgeSync. In most cases, you will create EventPaks that contain ALL of the triggers, events, reports, charts, scripts or web service calls for one or more applications in KnowledgeSync.

You can, however, create EventPaks with only one or two objects within them. To do this, you would:

Create a new application in KnowledgeSync called (for example) "Transfer EventPak". This will store the objects that you wish to put into your EventPak.
Copy-and-paste the desired triggers, events, et cetera from the application in which they currently reside to the application you just created.
Build your EventPak, including only those objects from the "Transfer EventPak" application.

13.2 Accessing EventPak Functions

Eventpak functions are located within the Event Designer option.

13.3 Vendor Identification

You have the option to brand an EventPak with your identification, to allow the recipients of the EventPak to know from whom they came and whom to contact for assistance. To enter your vendor identification, follow these steps:

Within the Event Designer, click on "EventPak" and then on the "Vendor Identification" option:

EVENTPAK VENDOR IDENTIFICATION

Fill in as many of these tabs as appropriate; the "General" tab lets you specify overall name and address information (along with notes about installing the EventPak), and the other tabs allow you to specify contact information for the various departments of your organization.

When finished entering this information, click on "Save".

13.4 Building an EventPak

To build an EventPak, follow these steps:

From the Event Designer option, select "EventPak" and then the "Builder" option:
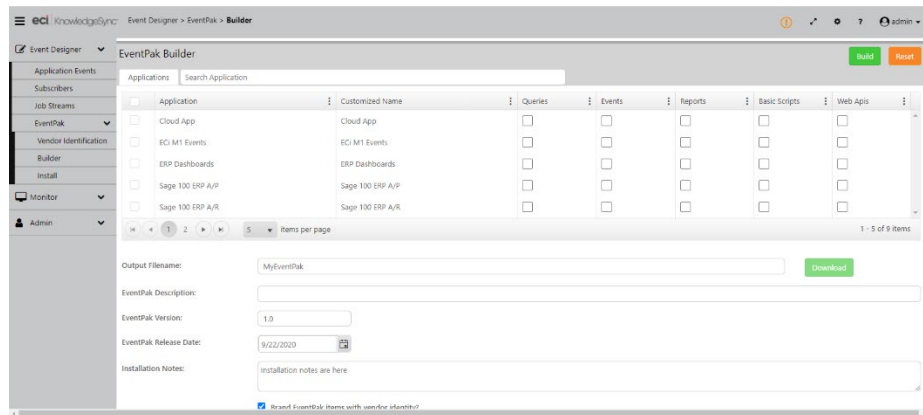
EVENTPAK "BUILDER" WINDOW

Note that it's best to display this option in full-window mode, or else some components on the window might not be readily visible.

The grid at the top of the EventPak Builder window displays a list of all of the applications that are currently defined in KnowledgeSync.

Place a checkmark in the "Include" box for any applications whose queries, events, reports, VB scripts and web APIs you wish to include.

For each application you select, place a checkmark in the columns that indicate which elements of event data you wish to include in the EventPak. You may choose from queries, events, reports, VB scripts and web APIs.

Once you have selected the applications, fill in the following fields:

The name and location of the EventPak output file. Eventpaks require a file suffix of ".epk".
The description of the EventPak.
The EventPak version number. This is very useful if you create updates to an EventPak, as it allows a recipient to tell the versions apart.
The date that the EventPak was created/released.
The EventPak's installation notes. This will be displayed when a recipient prepares to install an EventPak.
Whether to "brand" an EventPak with the "vendor identification" details you have entered. Vendor information will appear in the "Help" menu under the heading "About EventPak Vendor."
Click on the "Build" button to create your EventPak.

Eventpak files tend to be quite small and are ideal for distributing via email.

13.5 Installing an EventPak

To install an EventPak, follow these steps:

Under the Event Designer, select "Eventpak" and then "Install". You will be prompted to "Choose File" – the name and location of the EventPak you wish to install.
Once you have selected the eventpak and have clicked on the blue "Upload" button, you will see the following window:
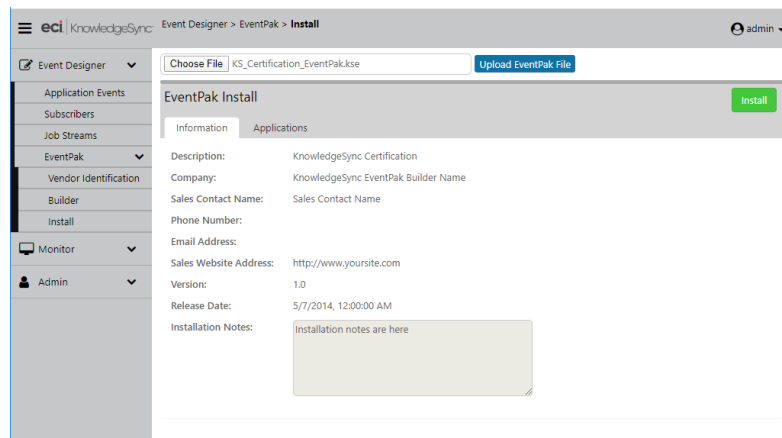
EVENTPAK INSTALLATION WINDOW

The "Information" tab contains information about the EventPak, who created it, when it was created, and any specific installation notes.

The "Applications" tab shows you (and lets you choose from) a list of the applications that are included in the EventPak. The column titled "Destination Type" controls how and where the EventPak will be installed:

- o Install it as a **new** application in KnowledgeSync.
- o **Add** its data to an existing application in KnowledgeSync

*KnowledgeSync will **not** overwrite existing queries or events but will prompt you regarding duplicate report definitions, web service APIs, and VB scripts.*

(If you add the contents of an EventPak to an application that already has triggers, reports, or events of that name, you will end up with **two** objects with the same name.)

Click on the "Install" button when you are ready to install the EventPak.

After installing the EventPak, go into the "Properties" option for that EventPak and specify the connection settings.

Eventpaks omit "connection" details (such as the Data Provider); thus, after you install an EventPak you must specify the connection data.

14 Utilities

14.1 External Message Editing

KnowledgeSync includes a way to enable users to edit and modify an event's alert message content without accessing the KnowledgeSync application. This function is referred to as "External Message Editing".

Instead of storing an event's alert message content within the "Email", "Text" or "Webcast" tabs of an event, you may store the event's message content within a field of a database table. After you do this,go into an event's "Email", "Text" or "Webcast" tabs, and refer to the name of the database <u>field</u> that contains the message content.

When the corresponding event is triggered, KnowledgeSync will notice that the contents of the alert message refer to a database field name. And, when KnowledgeSync translates that field, it will see that this field's contents consist not of hard-coded text, but rather a combination of hard-coded text and other database field variables.

KnowledgeSync will then translate those contents into the corresponding alert message text.

14.1.1 Create External Message Table

To use External Message Editing, you first need to create a table that will house the editable message format.

It does not matter what this table is called, or where it is stored, but it is suggested that if a client is using KnowledgeSync to monitor an application called "Target" that you create a new table in the "Target" database and call that table something like "KS Message." (You may use multiple tables that contain editable message formats for events.)

14.1.2 Create Field(s) for Message Formats

Since the new table will store the editable message format for an event, you need to create one or more data fields (columns) within the table that are going to hold the message content.

For example, if you wish to allow users to externally edit the email message text for a KnowledgeSync event, you might wish to create a field called something like "Email_Data" in the table you have created.

The columns that will hold the message formatting instructions must be "character" or "memo" type and must be capable of storing a large amount of alphanumeric data.

14.1.3 Create Access for Editing the Table

The next step is to identify how users will edit and/or customize the contents of the table created in the previous steps.

The method by which you allow users to edit this content is up to you; if you wish to allow non-technical users to perform External Message Editing, it is suggested that you create a friendly user interface that allows users to easily do this.

If, on the other hand, only programmers are going to be editing KnowledgeSync's messages externally, you can let programmers use any number of database access tools to edit the table's contents.

14.1.4 What Should the Table's Contents Look Like?

Consider the scenario where you've created a table called "KS Message" and you have a field within it called "Email_Data".

The contents of the "Email_Data" field would be the same as the contents of the "Email" tab for an event. For example, if you wished to create a message that sends a "thank you for ordering" message, the contents of the "Email_Data" field might look like the following:

Hello, {FIRSTNAME}:

Thank you very much for doing business with Acme Corporation. We have processed the following order for you:

Description: {DESCRIPTION}
Order Amount: {ACTUALAMOUNT}

Should you have any questions about this order, please feel free to contact your salesperson, {ACCT_MGR} via phone at 800-850-8055, or via electronic mail at {ACCT_MGR_EMAIL}.

Thank you again for doing business with us.

Acme Corporation
902 Hines Point
Vineyard Haven, MA 02568

Note the following about the contents of this column:

- Any of the database fields referred to in this column must also exist within the trigger that is used by the corresponding event.
- You may include both hard-coded text and database field values
- You may include date substitution variables
- You may include {BEGIN*REPEAT} and {END*REPEAT}.
- You may include {BEGIN*HTML} and {END*HTML}.
- You may include any of KnowledgeSync's field formatting commands.

14.1.5 Modify Trigger (Query) to Retrieve External Data

To configure an event to send an alert using data defined within an external table, you must ensure that the event's trigger retrieves not only all the necessary message data that is referenced within the external table, but that it retrieves data from the external table as well. Follow these steps (which use a query to retrieve database fields):

Open the event's query, go to the "Columns" tab, and add the table that contains the external message data. (You do not need to link the external message data table to the existing table for the query.)

Still on the "Columns" tab, add the column from the external message data table that contains the alert message text.

If your external message data table contains more than one record (e.g., if it contains multiple alert message formats), you must include filtering in your query to ensure that only a single record from that table is selected.

Query "Preview" mode will display the actual contents of the selected column; it will not show you the translated values of the database field values that are referenced within that external message table field.

14.1.6 Edit the Event to Reference the External Data

The final step in configuring an event to retrieve message data from the external table is to modify the event to refer to the data field in the external data table. Follow these steps:

- Open the Event you wish to use the external message data.
- Go to the appropriate message deliverable tab (i.e., "Email").
- In the "Message Body", go to the list of "Available Values for Use in Message Content" and select the field containing the external message data.

Note that an event may refer to multiple external data fields in its alert configuration. For example, your email message body could appear as follows:

{EMAIL_INTRO}
{EMAIL_BODY}
{EMAIL_CLOSING}