

CAPLIN

Caplin Xaqua 1.0

Alerts Overview And Concepts

August 2011

CONFIDENTIAL

Contents

1	Preface.....	1
1.1	What this document contains.....	1
	About Caplin document formats	1
1.2	Who should read this document.....	1
1.3	Related documents.....	2
1.4	Typographical conventions.....	3
1.5	Feedback.....	3
1.6	Acknowledgments.....	3
2	What are Alerts?	4
3	Alert features.....	8
4	Alerts architecture.....	11
5	Alert processing.....	13
6	Sending Alerts from a DataSource application.....	18
7	Configuring Caplin Xaqua for Alerts.....	19
8	Glossary of terms and acronyms.....	21

1 Preface

1.1 What this document contains

This document explains the concept of Alerts in trading systems, and gives an overview of how they work in Caplin Xaqua and Caplin Trader.

Tip: If you wish to add Alerts capability to existing applications or new applications based on the earlier Caplin Trader releases 2.0 or 2.1, please contact Caplin Support for advice and instructions on installing the required upgrade.

About Caplin document formats

This document is supplied in three formats:

- ◆ Portable document format (*.PDF* file), which you can read on-line using a suitable PDF reader such as Adobe Reader®. This version of the document is formatted as a printable manual; you can print it from the PDF reader.
- ◆ Web pages (*.HTML* files), which you can read on-line using a web browser. To read the web version of the document, navigate to the *HTMLDoc* folder and open the file *index.html*.
- ◆ Microsoft HTML Help (*.CHM* file), which is an HTML format contained in a single file. To read a *.CHM* file just open it – no web browser is needed.

For the best reading experience

On the machine where your browser or PDF reader runs, install the following Microsoft Windows® fonts: Arial, Courier New, Times New Roman, Tahoma. You must have a suitable Microsoft license to use these fonts.

Restrictions on viewing *.CHM* files

You can only read *.CHM* files from Microsoft Windows.

Microsoft Windows security restrictions may prevent you from viewing the content of *.CHM* files that are located on network drives. To fix this either copy the file to a local hard drive on your PC (for example the Desktop), or ask your System Administrator to grant access to the file across the network. For more information see the Microsoft knowledge base article at <http://support.microsoft.com/kb/896054/>.

1.2 Who should read this document

This document is intended for anyone who needs to understand how Alerts work in Caplin Xaqua. It is particularly relevant to:

- ◆ Technical Managers
- ◆ System Architects
- ◆ Software Developers

1.3 Related documents

- ◆ **Caplin Trader API Reference**

Documents the JavaScript API that allows developers to extend Caplin Trader by writing custom JavaScript code. The package **caplin.alerts** provides an API that allows Alerts capability to be built into Caplin Trader applications.

- ◆ **DataSource for C Xaqua Extensions API Reference**

Contains API reference documentation for the **Alerts module** that enables a C DataSource application to send Alert Notifications originating from a bank's systems to Caplin Xaqua clients.

- ◆ **DataSource for Java API Reference**

The API reference documentation for the Java DataSource SDK.

The package **com.caplin.datasource.extensions.alerts** enables a Java DataSource application to send Alert Notifications that originate from a bank's systems to Caplin Xaqua clients.

- ◆ **Caplin DataSource Overview**

A technical overview of Caplin DataSource.

- ◆ **DataSource For C Configuration Syntax Reference**

Describes the syntax of the language that is used to configure Caplin's DataSource For C product range, including the Caplin Liberator and Caplin Transformer components of Caplin Xaqua.

1.4 Typographical conventions

The following typographical conventions are used to identify particular elements within the text.

Type	Uses
aMethod	Function or method name
<i>aParameter</i>	Parameter or variable name
<i>/AFolder/Afile.txt</i>	File names, folders and directories
<div>Some code;</div>	Program output and code examples
The value=10 attribute is...	Code fragment in line with normal text
Some text in a dialog box	Dialog box output
Something typed in	User input – things you type at the computer keyboard
Glossary term	Items that appear in the “Glossary of terms and acronyms”
XYZ Product Overview	Document name
◆	Information bullet point
■	Action bullet point – an action you should perform

Note: Important Notes are enclosed within a box like this.
Please pay particular attention to these points to ensure proper configuration and operation of the solution.

Tip: Useful information is enclosed within a box like this.
Use these points to find out where to get more help on a topic.

Information about the applicability of a section is enclosed in a box like this.
For example: “This section only applies to version 1.3 of the product.”

1.5 Feedback

Customer feedback can only improve the quality of our product documentation, and we would welcome any comments, criticisms or suggestions you may have regarding this document.

Visit our feedback web page at <https://support.caplin.com/documentfeedback/>.

1.6 Acknowledgments

Adobe® Reader is a registered trademark of Adobe Systems Incorporated in the United States and/or other countries.

Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

2 What are Alerts?

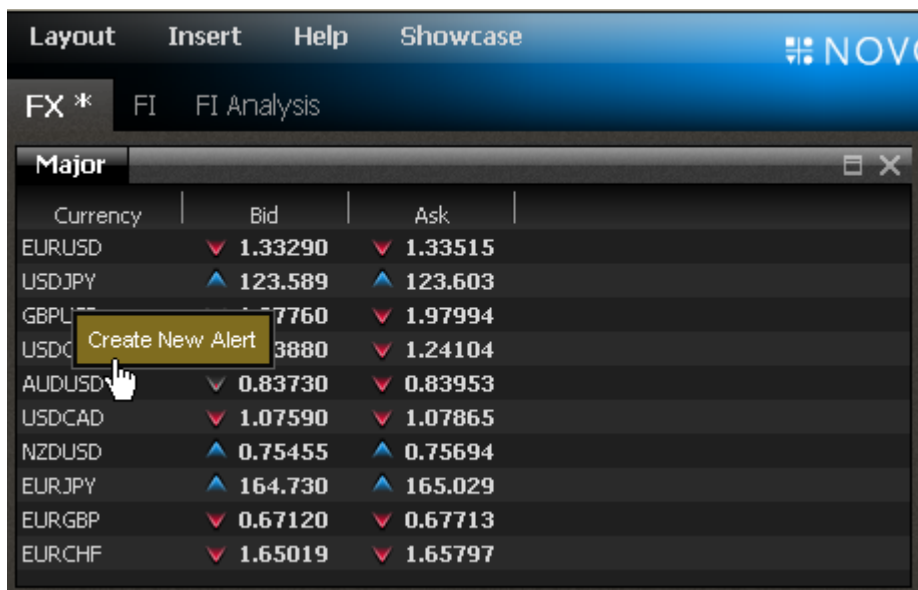
Financial traders often make the decision to trade particular instruments based on some dynamically changing attribute of those instruments. For example, an FX trader may decide to buy a particular currency only when its ask price goes below a certain value.

Caplin Xaqua's Alerts facility supports this workflow by automatically tracking changes in instrument attributes. The following example illustrates how this works. The pictures show a Caplin Xaqua client based on Caplin Trader.

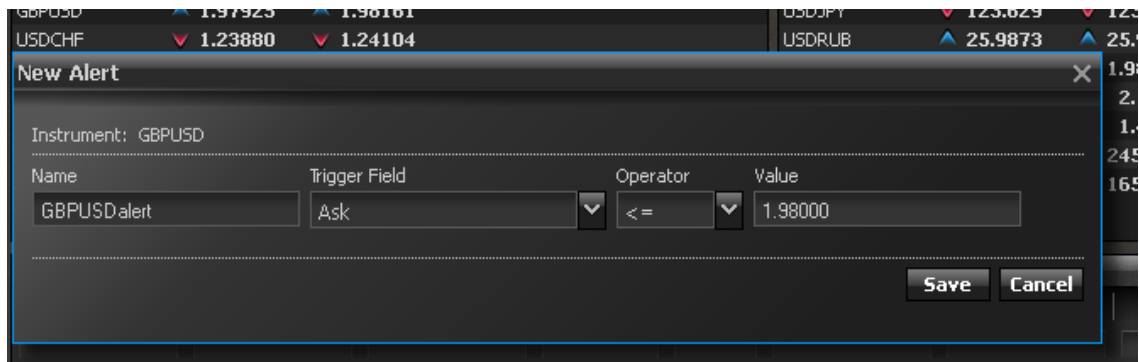
- Select an instrument; in this example it is GBPUSD



- Right click on the instrument name and click on the pop-up menu to create a new **Alert**.

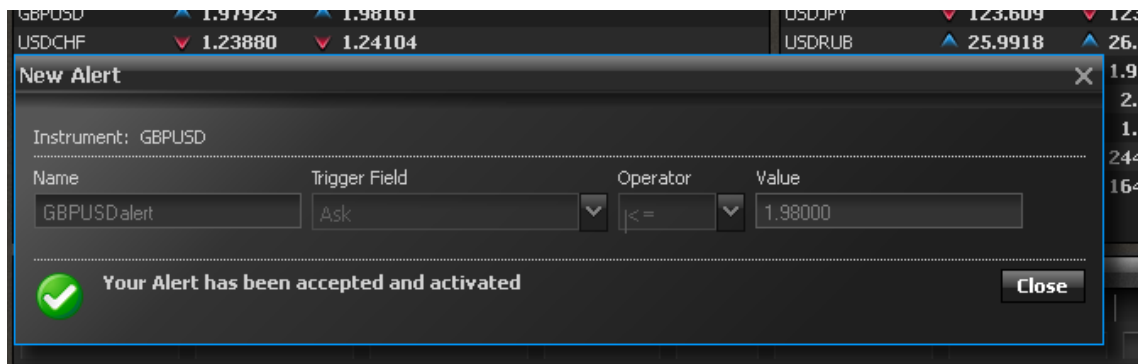


- In the New Alert dialog box, name the Alert and enter the **Alert Condition**: Ask <= 1.98000.

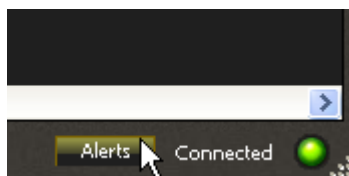


The combination of the instrument name and Alert Condition is called an **Alert Trigger**. This is the information that Caplin Xaqua monitors to determine when the end-user should be told that the alert has fired.

- Click on Save. The Alert is accepted and activated.



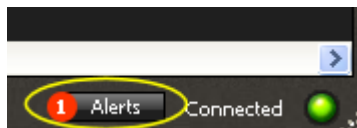
- Select the Alerts button to see the list of **Active Alerts**.



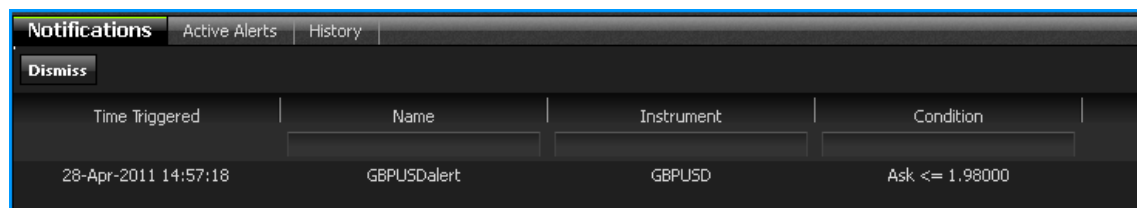
The **Alerts Manager** shows the list of Active Alerts in the **Active Alerts Grid**. Here the Grid shows just a single Alert; the one that has just been set up.



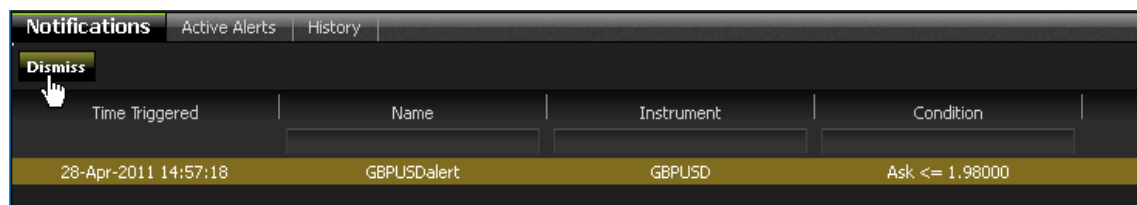
Some time later the condition specified in the Alert is met—the Ask price of GBPUSD drops to 1.98000 or less. The client is notified through a counter on the Alerts button (outlined in yellow in the following picture):



The **Notifications Grid** of the Alerts Manager shows the details of the notified Alert.



- Select the notified Alert and click on the Dismiss button to dismiss it.



The **Notification History Grid** of the Alerts Manager shows the details of all the Alerts that have been notified. Here the Grid shows just a single Alert; the one that has just been dismissed.

Notifications	Active Alerts	History	
Time Triggered	Name	Instrument	Condition
28-Apr-2011 14:57:18	GBPUSDalert	GBPUSD	Ask <= 1.98000

3 Alert features

The features provided in the Alerts facility are listed in the tables below, followed by a note of the restrictions that apply when this document was published.

At the time of publication, only Caplin Trader clients can use the Alerts facility, by means of the **Alerts API** provided with Caplin Trader. It is anticipated that future releases of Caplin's trading software will include client support for Alert handling in other technologies (such as Microsoft .NET)

Tip: Caplin Trader includes an **Alerts blade** that is an example implementation of Alert handling in a Caplin Trader application. The screen shots shown in [What are Alerts?](#)⁴ are based on the Alerts blade.

Key to the following tables:



Feature available by end Q2 2011.






Feature may be available in future releases of Caplin Xaqua and Caplin Trader (after Q2 2011).

The "Technology Area" column in the table shows where the feature is implemented: Caplin Xaqua, Caplin Trader (Alerts blade and/or Alerts API), or both.





User workflow features

Feature	Technology Area	Availability
Add and delete Alert Triggers, and view active Alerts.	Caplin Xaqua Caplin Trader	
Receive, view and dismiss Alert Notifications.	Caplin Trader Caplin Xaqua	
View and delete Alerts history in a Notifications History Grid.	Caplin Trader Caplin Xaqua	
Alert Notifications appear in pop-ups.	Caplin Trader	
Launch a trade ticket from the Alert Notification pop-up.	Caplin Trader	
Edit a previously defined Alert Trigger.	Caplin Trader	
Create a new Alert Trigger by editing the details of an Alert Notification in an Alert Notification popup.	Caplin Trader	



Alert Conditions

Feature	Technology Area	Availability
Alert Triggers with one condition (for example Ask <= 1.98000).	Caplin Xaqua Caplin Trader	
Alert Condition monitoring based on the end-user's price tier.	Caplin Xaqua Caplin Trader	
Alert Triggers with two conditions (for example Ask <= 1.98000 AND Bid >= 1.97700).	Caplin Xaqua Caplin Trader	


Notifications

Feature	Technology Area	Availability
Notifications are raised for any Alert Conditions that are met at the time an end-user logs in.	Caplin Xaqua Caplin Trader	
Notifications originating in the bank's systems can be sent to end-users via a dedicated DataSource adapter.	Caplin Xaqua	
Alert Notifications appear in pop-ups.	Caplin Xaqua	
Alert Notifications are shown against the relevant instruments in Instrument Grids.	Caplin Xaqua	

Persistence

Feature	Technology Area	Availability
An end-user's Active Alerts are persisted on the Caplin Trader database across the user's Caplin Trader sessions. The alerts are monitored when the user logs in again.	Caplin Trader	
Server-side persistence: An end-user's Active Alerts are persisted in Caplin Xaqua across their Caplin Trader sessions. This allows the user to be given an Alert Notification whose condition was met during the time they were logged out of Caplin Trader.	Caplin Xaqua Caplin Trader	

Resilience and availability

Feature	Technology Area	Availability
Alerts continue to be monitored when Caplin Xaqua components (Liberator, Transformer, and DataSource adapters) fail over to back-up components. In the rare event of all Caplin Xaqua components going down, alert monitoring is resumed as soon as the components become available again.	Caplin Xaqua Caplin Trader	

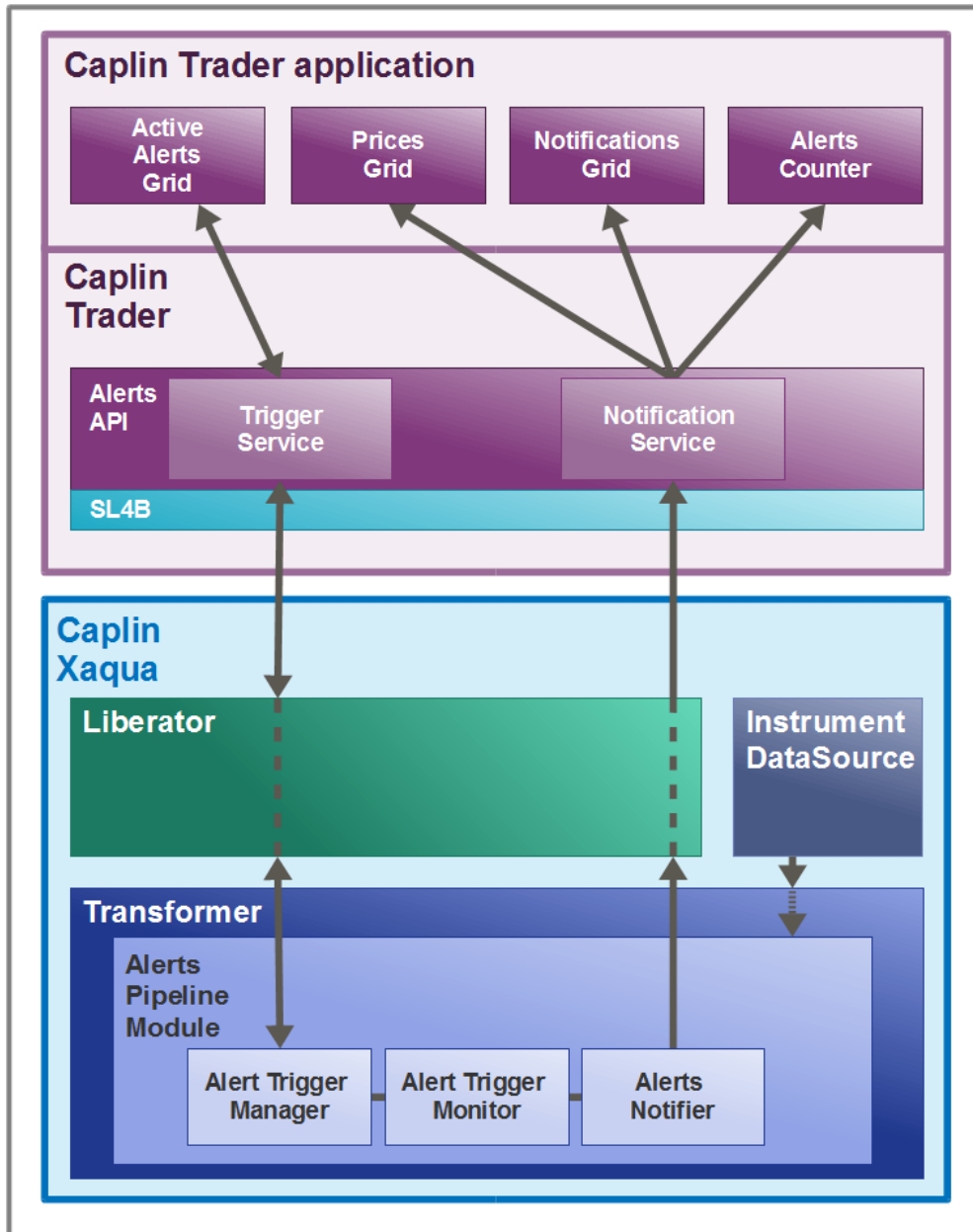
Restrictions

At the time of publication the Alerts facility has the following restrictions:

- ◆ Alert Conditions can only refer to numeric fields.
For example, `Bond_rating > "BBB"` is not a valid condition.
- ◆ An Alert Trigger can only be applied to a single item (instrument), not multiple items.
For example, to monitor the price of the FX currency pairs GBPUSD and GBPCAD, a separate Alert Trigger must be defined for each pair.
- ◆ Notifications are only generated when the end-user is logged in; when the end-user is logged out no notifications are generated. This restriction may be removed in future releases of the Alerts facility (see the "Server-side persistence" feature in the "Persistence" features category).
- ◆ Notifications do not persist across end-user sessions. However, Active Alerts are persisted (see the "Persistence" category in the features table).
- ◆ Changes to the layout of the Alerts Manager in the Caplin Trader Alerts Blade do not persist across end-user sessions.

4 Alerts architecture

The Alert facility is implemented through Caplin Xaqua components interacting with a **Caplin Xaqua client**. The following diagram shows this architecture where the client is a **Caplin Trader application**.



Alerts Architecture

In the diagram, the **Caplin Trader application** has **display components** for an Active Alerts Grid, Price Grids, a Notification Grid, and an Alerts Counter, as shown in the screen shots in [What are Alerts?](#)⁴
The Alerts Counter is part of the Alerts button:



The **Caplin Trader Alerts API** allows any display component to create Alert Triggers and receive Alert Notifications. The API consists of a Trigger Service and a Notification Service that collectively manage the communication of Alert information between the Caplin Trader application and Caplin Xaqua.

The Alerts API communicates with Caplin Xaqua via **StreamLink for Browsers (SL4B)**.

The **Caplin Liberator** component of Caplin Xaqua routes Alert related messages between Caplin Trader and Caplin Transformer, mapping the subjects of these messages to meet the requirements of Transformer.

The **Caplin Transformer** component of Caplin Xaqua contains an **Alerts Pipeline Module** that manages Alerts on behalf of client applications. This pipeline module consists of the following sub-modules:

◆ **Alert Trigger Manager**

The Alert Trigger Manager manages the Alert Triggers that have been created by end-users and sent by Caplin Trader's Trigger Service. The Alerts are held in a dedicated **container** for each user.

◆ **Alert Trigger Monitor**

The Alert Trigger Monitor checks whether the condition of each Alert Trigger has been met. It informs the Alert Trigger Manager and Alerts Notifier of such matured Alerts.

◆ **Alert Notifier**

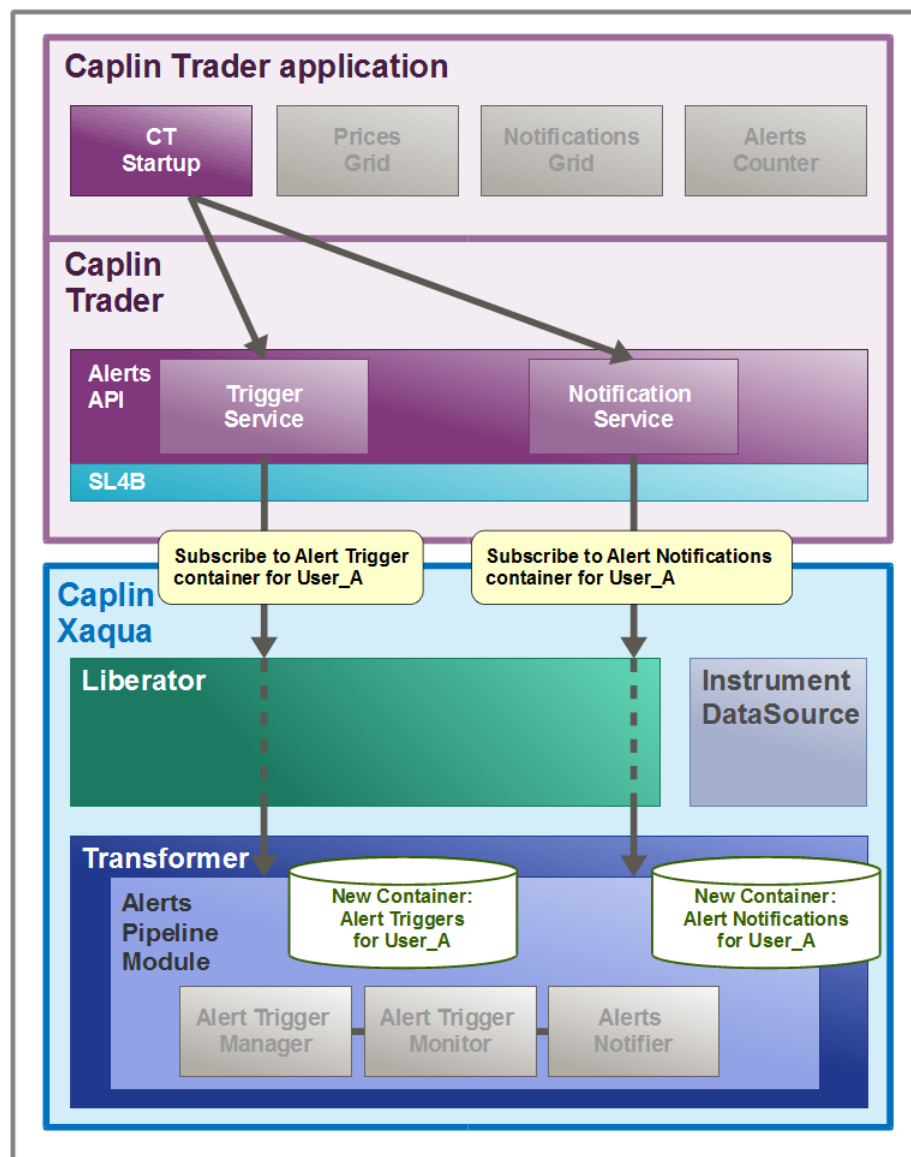
The Alert Notifier creates an Alert Notification message for each Alert Trigger passed to it, and sends the message to Caplin Trader's Notification Service via Liberator and SL4B.

Transformer must be connected to one or more **Instrument DataSources**. These are **DataSource adapters** that supply the static information and real-time updates (for example, price updates) about the instruments that can be traded. Liberator subscribes to this instrument data on behalf of Caplin Trader clients, via the Transformer, so that the data can be displayed on the client in grids and other display components. Transformer's Alerts Pipeline Module uses the instrument data to determine when Alert Conditions are met.

5 Alert processing

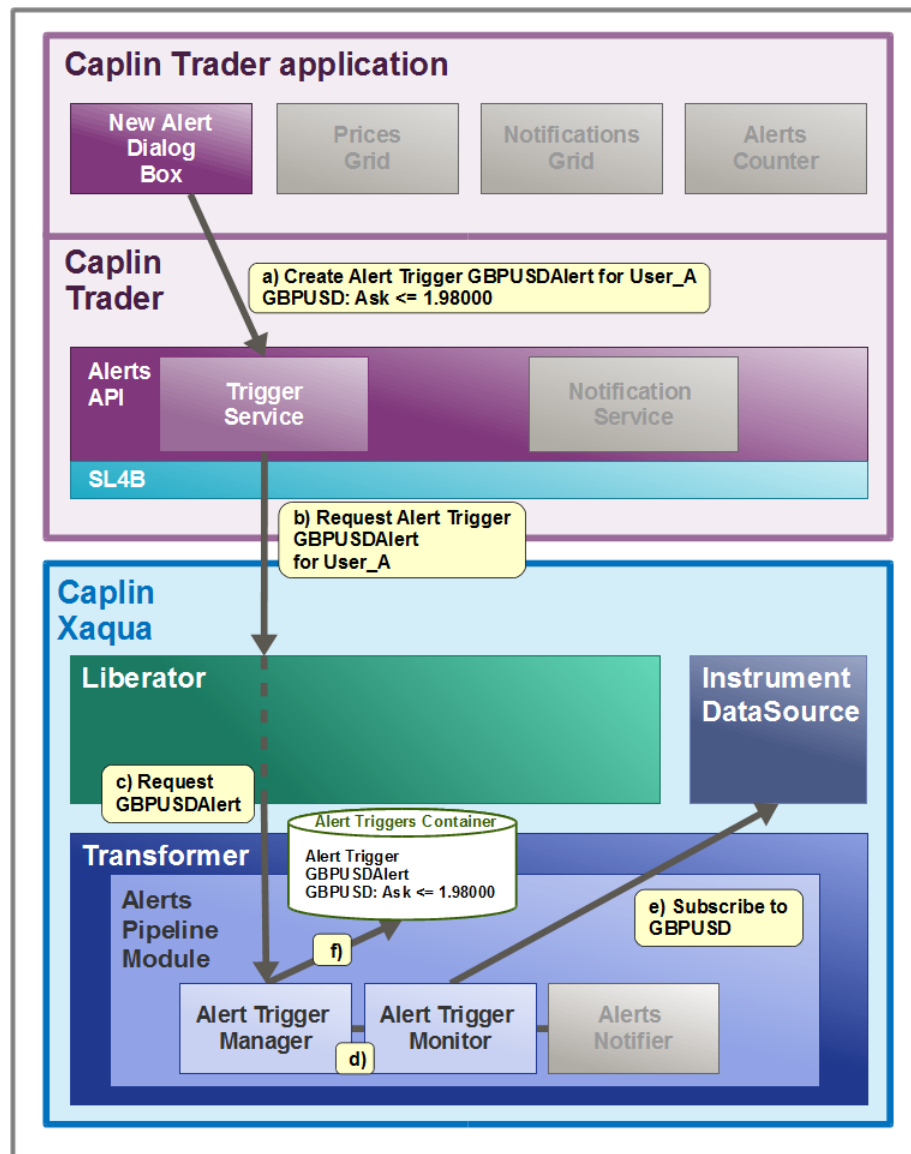
The following diagrams show the life cycle of an Alert as it is processed within Caplin Trader and Caplin Xaqua.

1. When the Caplin Trader application starts up, the Trigger Service subscribes to an Alert Trigger Container for the logged in end-user ('User_A' in the rest of this section), and the Notification Service subscribes to an Alert Notifications Container for that user. These containers are created on the Transformer and are used to manage the user's Alert Triggers and Alert Notifications. Changes to alert data are sent back to Caplin Trader as updates to the contents of these containers.



Alerts initialization

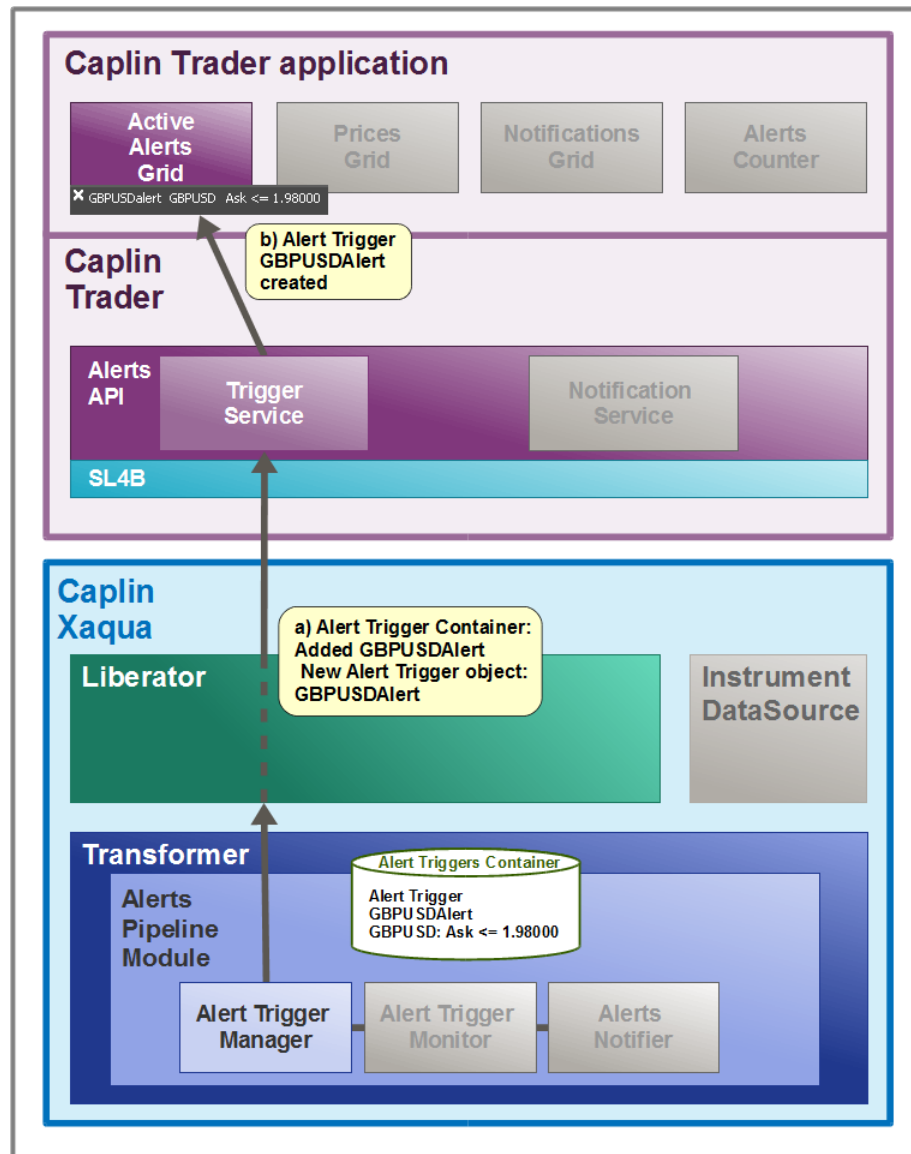
2. User_A creates an Alert called GBPUSDAAlert on an FX currency pair GBPUSD, with the Alert Condition $ASK \leq 1.98000$.



End-user creates an Alert Trigger

- a) In the Caplin Trader application the New Alert Dialog Box calls the Alerts API to request that the new Alert Trigger be created.
- b) The API's Trigger Service turns the request into an **RTTP** message and sends the message (via SL4B), as a contribution to the Liberator.
- c) Liberator passes the request on to Caplin Transformer, where it is handled by the Alert Trigger Manager in the Alerts Pipeline Module.
- d) The Alert Trigger Manager pushes the trigger to the Alert Trigger Monitor, so that it can start monitoring the trigger's condition.
- e) The Alert Trigger Monitor subscribes to the instrument that is to be monitored (GBPUSD) on the DataSource that supplies this instrument (if the Transformer is not already subscribed to it).

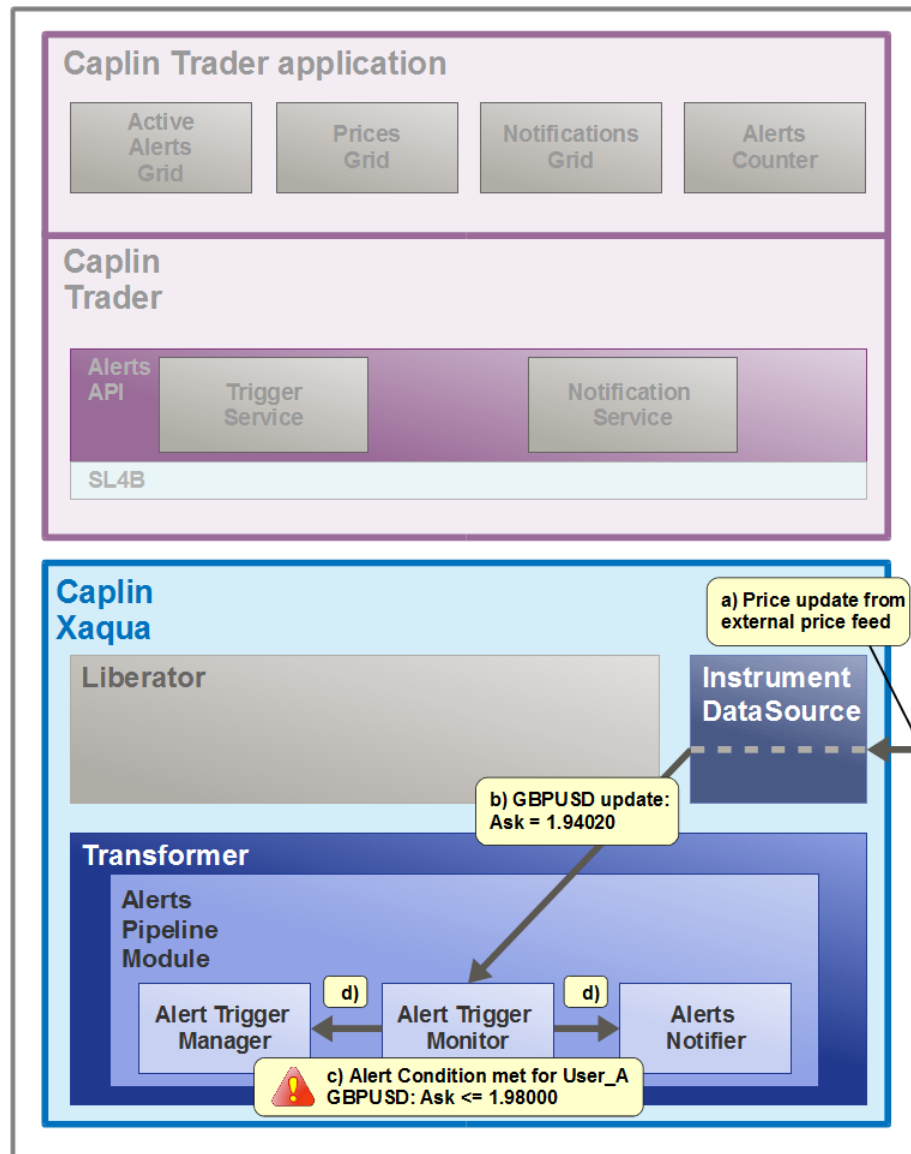
- f) The Alert Trigger Manager then creates an Alert Trigger object for this instrument, and puts a reference to the object in User_A's Alert Trigger Container.
3. The Transformer confirms that the Alert Trigger has been created:



Transformer confirms Alert Trigger created

- The Transformer's Alert Trigger Manager sends the Alert Trigger object and a corresponding Alert Trigger Container update back to Caplin Trader, via the Liberator. The Trigger Service receives a callback from StreamLink for Browsers that the Alert Trigger was added to the Alert Trigger Container; this confirms that the Transformer is now monitoring the trigger.
- The Trigger Service in Caplin Trader sends an 'Alert Trigger created' message to the Active Alerts Grid and the Grid is populated with the Alert Trigger information.

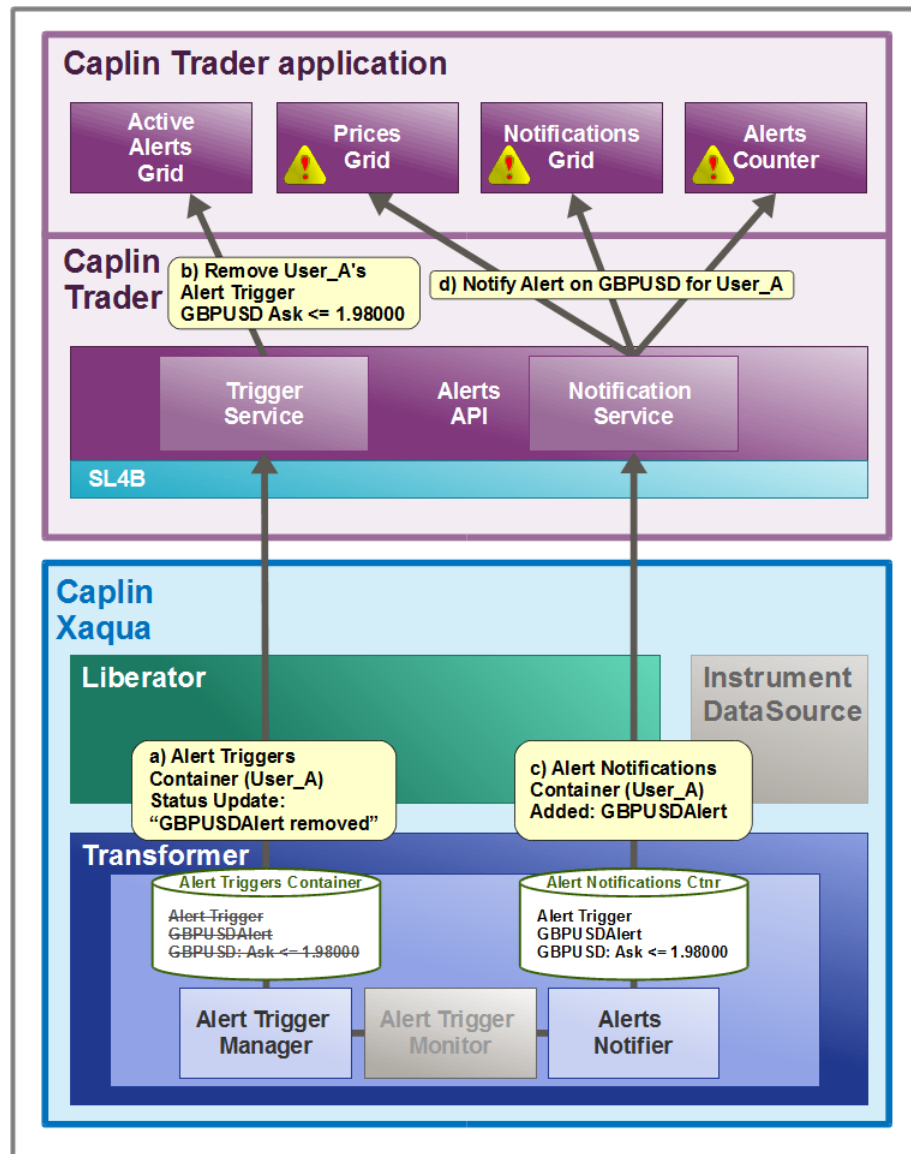
4. Sometime later the Alert Condition is met:



Alert Condition is met

- A change to the Ask price of GBPUSD comes in to the Instrument DataSource from an external price feed.
- When the Caplin Trader application originally requested the GBPUSD instrument to display in its Prices Grid, the Transformer subscribed to the instrument GBPUSD, so the DataSource now sends the price update on to the Transformer.
- The Transformer's Alert Trigger Monitor checks the update and determines that the Ask price is now below the price specified in User_A's Alert Trigger – the Alert Condition has been met.
- The Alert Trigger Monitor calls the Alert Trigger Manager and the Alerts Notifier.

5. An Alert Notification is raised.



Alert Notification is raised

- The Alert Trigger Manager sends a status update for User_A's Alert Triggers Container to the Caplin Trader Trigger Service, via Liberator.
- The Trigger Service sends a "remove trigger" notification to the Active Alerts Grid, which can then remove the Alert Trigger GBPUSDAsk from the grid.
- The Alerts Notifier sends the Caplin Trader Notification Service an update to the Alert Notifications Container, via Liberator.
- The Notification Service sends an Alert Notification for the GBPUSDAsk Alert Trigger to all interested display components – in this example these components are: the Prices Grid (which flags the Alert against the instrument in the grid), the Notifications Grid (which puts the Alert Trigger in the Grid), and the Alerts Counter (which increments by one the number of Alerts received).

6 Sending Alerts from a DataSource application

You can also send Alert Notifications to Caplin Xaqua clients directly from a DataSource application. Such Notifications are not associated with any user-defined Alert Triggers and they typically originate in the bank's systems – the DataSource application receives information from a bank system and passes it on to the client as an Alert Notification.

- To send Alert Notifications from a C DataSource application, use the **Alerts module** of the **DataSource Xaqua Extensions API**.
- To send Alert Notifications from a Java DataSource application, use the package **com.caplin.datasource.extensions.alerts** (see the **DataSource for Java API Reference**).

7 Configuring Caplin Xaqua for Alerts

Tip: The Caplin Trader 2.2.x installation kit includes a Caplin Transformer and Caplin Liberator that are configured to provide Alerts functionality “out of the box”. You do not need to change this configuration when installing Caplin Trader for evaluation purposes only. However, if you are implementing a deployable Caplin Trader application, you may need to change the configuration, as explained below.

Tip: If you wish to add Alerts capability to existing applications or new applications based on the earlier Caplin Trader releases 2.0 or 2.1, please contact Caplin Support for advice and instructions on installing the required upgrade.

To implement Alerts handling in your Caplin Xaqua based trading system, you need:

- ◆ Caplin Transformer with the Alerts Pipeline Module installed.
- ◆ Caplin Liberator.
- ◆ One or more Instrument DataSources (**DataSource Adapters**).

The DataSource configuration for Liberator and Transformer must contain object definitions, object maps and data service definitions that support the Alerts messaging. The Alerts Pipeline Module has an associated Alerts Configuration file, *rttpe-alerts.conf*, that provides these definitions. This file should be added to the Liberator configuration file *etc/rttpe.conf* using an **include-file** directive (the document **DataSource For C Configuration Syntax Reference** explains how to use **include-file**).

You may have already made changes to your Liberator configuration to reflect the naming conventions used for your Caplin Xaqua installation. In this case, you may need to modify *rttpe-alerts.conf* to change the name (**label**) of the Transformer in the **add-data-service** definition for the Alerts service, so that it matches the label defined in the Liberator's **add-peer** configuration defining the Transformer:

Liberator configuration *rttpe.conf*: add-peer for Transformer

```
...
# transformer
add-peer
  remote-id           20
  remote-type         active
  remote-name         Transformer-A
  label               Transformer-A
end-peer
...
```

Transformer label in *rttd-alerts.conf*

```
...
add-data-service
    service-name          alerts
    required-state        up
    include-pattern        ^/ALERTS
    include-pattern        ^/NOTIFICATIONS

    add-source-group
        required
        add-priority
# Modify this label as required ->
        label transformer
# For example:
        label Transformer-A
    end-priority
    end-source-group
end-data-service
...
```

8 Glossary of terms and acronyms

This section contains a glossary of terms, abbreviations, and acronyms relating to Alerts in Caplin Xaqua and Caplin Trader.

Term	Definition
Active Alert	An Alert Trigger that is being monitored by Caplin Xaqua .
Active Alerts Grid	A Caplin Trader Grid showing on the client all Active Alerts set up by the end-user.
Alert	A request made to Caplin Xaqua by a Trader to be notified when an instrument meets a specified set of criteria (the Alert Condition), and the corresponding response. The content of the request is an Alert Trigger , and the request is made through a Caplin Xaqua client.
Alerts Blade	A blade that consists of an example implementation of the Alerts functionality available to Caplin Trader .
Alerts Manager	A pop-up panel in the Alerts blade that allows end-users to view and manage Alerts . It contains tabbed grids showing Active Alerts, Alerts that have been notified, and an Alert History list. For a picture of the Alerts Manager, see What are Alerts? ^[4]
Alert Trigger	An instrument and associated Alert Condition that are monitored to judge whether Caplin Xaqua needs to notify the end-user that the alert has fired.
Alert Notification	A message to the client reporting that the Alert Condition for an Alert Trigger has been met.
Alert Condition	The criteria that, when met, cause an Alert Notification to be sent for a particular instrument.
Blade	A business component that provides domain specific functionality in a Caplin Trader application . Each Caplin Trader blade implements a small, well-defined set of closely related functions.
Caplin Liberator	Caplin Liberator is a real-time financial internet hub that delivers trade messages and market data to and from subscribers over any network.
Caplin Trader	A web application framework for constructing browser-based financial trading applications (Caplin Trader applications).
Caplin Trader application	A Caplin Xaqua client that has been built using Caplin Trader .
Caplin Transformer	Caplin Transformer is an event-driven real-time business rules engine.
Caplin Xaqua	A framework for building single-dealer platforms that enables banks to deliver multi-product trading direct to client desktops.
Caplin Xaqua client	A client desktop or web application that interfaces with Caplin Xaqua to deliver multi-product trading to end users. Also see Caplin Trader application .
Client	In this document, “client” is short for Caplin Xaqua Client .
Condition	In this document, “condition” is short for Alert Condition .
Container	In Caplin Xaqua a container object holds a set of references to other objects; this allows related objects to be grouped together.

Term	Definition
	For more information see the Caplin DataSource Overview .
DataSource adapter	A DataSource application that integrates with an external (non-Caplin) system, exchanging data and/or messages with that system.
DataSource application	A Caplin Xaqua application that uses the Caplin DataSource APIs to communicate with other Caplin Xaqua applications via the DataSource protocol.
	For more information, see the Caplin DataSource Overview .
Display component	A GUI component of Caplin Trader that can be rendered in a page on the screen. The term also refers to the JavaScript code that generates the component and handles its user interaction.
Grid	A Caplin Trader display component that displays data in a tabular format.
GUI	<u>G</u> raphical <u>U</u> ser <u>I</u> nterface.
Notification	In this document, “notification” is short for Alert Notification .
Notification History Grid	A Caplin Trader Grid on the client that shows all Alert Notifications received during the current user session.
Notifications Grid	A Caplin Trader Grid on the client showing all Alert Notifications received during the current user session that have not been dismissed by the end-user.
RTTP	<u>R</u> ea <u>L</u> <u>T</u> ime <u>T</u> ext <u>P</u> rotocol. Caplin's protocol for streaming real-time financial data from Caplin Liberator servers to client applications, and for transmitting trade messages and other messages between clients and Liberator in both directions.
SL4B	<u>S</u> tr <u>e</u> a <u>m</u> <u>L</u> i <u>n</u> k for (4) <u>B</u> rowsers
StreamLink	The StreamLink libraries connect client applications to Caplin Liberator via the RTTP protocol. They provide an object oriented API that gives access to RTTP functionality.
StreamLink for Browsers	StreamLink for Browsers is a JavaScript implementation of StreamLink that runs in Web browsers. It allows Caplin Trader applications to communicate with Caplin Liberator .
Trigger	In this document, “trigger” is short for Alert trigger.

Contact Us

Caplin Systems Ltd
Cutlers Court
115 Houndsditch
London EC3A 7BR
Telephone: +44 20 7826 9600
www.caplin.com

The information contained in this publication is subject to UK, US and international copyright laws and treaties and all rights are reserved. No part of this publication may be reproduced or transmitted in any form or by any means without the written authorization of an Officer of Caplin Systems Limited.

Various Caplin technologies described in this document are the subject of patent applications. All trademarks, company names, logos and service marks/names ("Marks") displayed in this publication are the property of Caplin or other third parties and may be registered trademarks. You are not permitted to use any Mark without the prior written consent of Caplin or the owner of that Mark.

This publication is provided "as is" without warranty of any kind, either express or implied, including, but not limited to, warranties of merchantability, fitness for a particular purpose, or non-infringement.

This publication could include technical inaccuracies or typographical errors and is subject to change without notice. Changes are periodically added to the information herein; these changes will be incorporated in new editions of this publication.

Caplin Systems Limited may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time.

This publication may contain links to third-party web sites; Caplin Systems Limited is not responsible for the content of such sites.