

BinX – Dynamic exploration of time series

CPSC 533
Course project

Lior Berry

Department of Computer Science
University of British Columbia
Vancouver BC Canada V6T 1Z4
berry@cs.ubc.ca

Abstract

Time series are a substantial data format, common to many fields of study and are becoming longer with the availability of new data accumulation technologies. One of the prominent time series analysis techniques is visualization. However, humans will often fail when analyzing a long time series, being overwhelmed by the amount of data involved and the different time scales it is encoded in. These often require multiple views, leading to disorientation or requiring the use of complex mathematical transformations (such as Fourier / Wavelet transforms). In this work I introduce BinX, an interactive tool, providing dynamic visualization and manipulation of long time series. The data is visualized via user controlled aggregation, augmented by several information visualization techniques. I believe that BinX provides simple, effective and flexible exploration methods for uncovering the underlying patterns in long time series.

1. Introduction

Time dependent data and time series are common to almost every field of study (science, engineering, medicine, business and more). Whenever a process is monitored or an activity is recorded, the result is a time series (e.g. daily stock quotes, hourly air temperature etc.). Therefore, an important part of understanding a time dependant process is analyzing the underlying time series. A time series has an independent dimension (time) and one or more dependent dimensions. The described work focused on time series taken at discrete time stamps $\{t_i\}$ with one or two time dependent dimensions. The time series can thus be defined as:

$S = \{(t_1, v_1), (t_2, v_2), \dots, (t_n, v_n)\}$, where $v_i = f(t_i) = (d_{i1}, d_{i2}, \dots, d_{ik})$. (k is the number of time dependent dimensions and each d_j is the value of the time dependent dimension j in time t)

Time series have been studied for a long time. Many analysis techniques have been developed, such as the use of the Fourier transform, wavelets and statistical operations. Still, visualizing the analyzed time series is a prominent analysis technique. The purpose of such visualizations is “to assist the human in detecting and validating characteristic properties of an unknown function f ” [1]. Some of the questions

that time series analysis should answer are: How often does a data element occur (Temporal texture)? What is the rate of change? Is there a pattern? Are two series / variables similar? Do any of the series match a pattern? When was something greatest/least?

Earlier time series visualizations were static and therefore limited in the amount of information displayed. As the amounts of information available for analysis grow and analyzed time series are becoming long, there is a need to effectively visualize long time series (both univariate and multivariate). This requires dynamic and more complex systems and is a current research topic.

Humans will often fail when analyzing a long time series, overwhelmed by the amount of data involved, which cannot fit the screen in a single view and also due to the different time scales the information is encoded in (e.g. days, months, years etc.). Such series often require multiple views (for different variables and for different time scales) leading to disorientation or requiring the use of mathematical transformations (such as Fourier / Wavelet transforms) that move the problem to the unintuitive frequency domain.

BinX was developed to provide dynamic visualization and manipulation of long time series. It assists a human in gaining better understanding of the data and underlying patterns, without requiring many views or applying complex transformations. The data is handled using aggregation techniques, where the data from several time intervals is aggregated and visualized as a single representative value (this process is also known as binning). Aggregation has been shown to be an effective method for dealing with large data sets [2],[10].

2. BinX

BinX facilitates dynamic exploration of time series. It provides interactive and smooth control of the aggregation level in which the series is displayed, quick navigation to selected time points (on and off screen) and clustering of the binned data. This section will describe the tool and its underlying Information visualization principles.

2.1 data

In the development of BinX, I have focused on visualizing currency exchange rates over time. The data comprises the daily USD exchange rate for several currencies over a period of 15 years (the data was collected using FXHistory [4]). Even In this medium size data set, each currency has more than 5000 data points for the analyzed period, making it nearly impossible to view all the data on a single screen. Although only this data set was used when developing BinX, the tool was developed to accommodate general time series, spreading along different periods.

2.2 tasks

The primary task BinX was designed to support is the understanding of a single currency exchange rate behavior along time. The task builds on the following subtasks, emanating from the “*Overview first, zoom and filter, then details-on-demand*” principle ([5]):

- Obtaining a global view in different detail levels (aggregation levels)
- Detecting time periods with interesting features, which are worth deeper exploration
- Comparing exchange rate across time periods (weeks, months, years) and finding similar periods.
- Finding global and local min / max.
- Monitoring a specific period in different aggregation levels

The secondary task supported by BinX is conducting a comparison between two currencies. This task requires the detection of a correlation, if there is one, between the two currencies (this could be a global correlation or in specific periods).

2.3 Time series visualization component

The principle component of BinX is the dynamic time series visualization component (DTVC). This is a self-contained component, presenting the time series with a controllable aggregation level, time axis scaling level and providing miscellaneous information on the binned data.

2.3.1 Controlling the aggregation level and time scale (Zooming)

Aggregation is used to reduce the amount of information displayed, so that the user can effectively explore the data. Several aggregation levels are required to allow full understanding. A common problem with static visualizations that use aggregation is the lack of clear connection between separate visualizations in different aggregation levels (often the axes and labels are changed, the span of the binned data is not always clear), forming a cognitive load for the user.

The DTVC visualization has a fixed bottom part, showing time marks and labels that serve as an anchor across aggregation levels. The top part visualizes the time series in the desired aggregation level with time marks connected to the static part, forming a dynamic trapeze-like shape (see Figure 1.a, 1.b). When narrowing the “trapeze” there is less room for information, resulting in an automatic increase of the aggregation level (**automatic aggregation** and its importance are discussed in [2]). The width of the trapeze clearly shows the level of aggregation, reducing cognitive load.

Another zooming mechanism controls the time scale or the horizontal spacing between data points. This mode of zooming is useful when the bin information is cluttered due to lack of space. Together with the navigation mechanism described later, these zooming options allow the user to get a high level view of the time series and then focus on the details of desired time periods, by bringing them to the center of display (focusing) and zooming in.

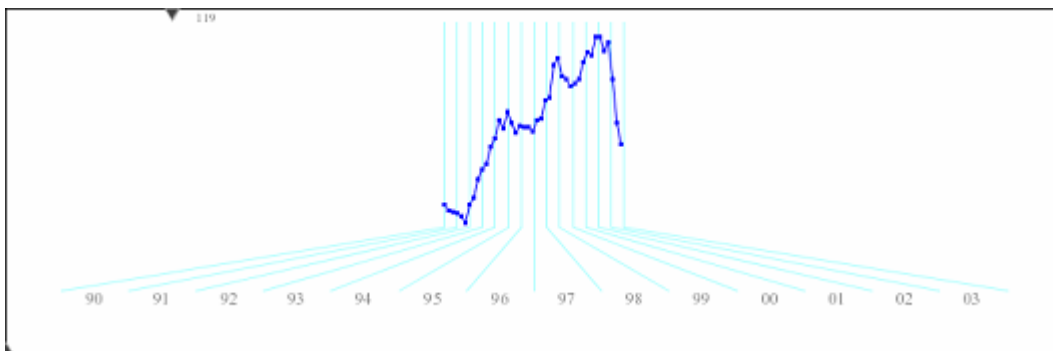


Figure 1.a The time series presented with an aggregation level of 119 (each 119 days are shown as a single dot), giving a rough overview for the user and reducing information from about 5000 points to 45

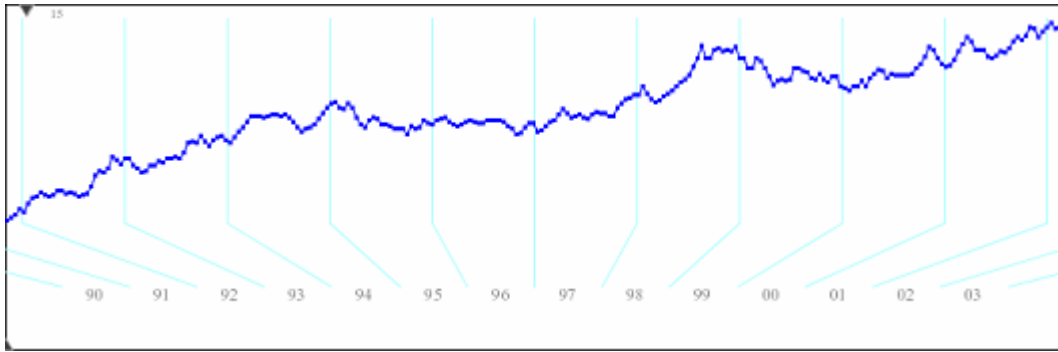


Figure 1.b The same time series after sliding the aggregation level knob to a finer aggregation level (21). Notice that the bottom part remains affixed, providing a constant context. Some of the information is off screen (e.g. the data for 2002-2003), yet the time marks (cyan) provide visual cues on that existence of such information.

2.3.2 Animation

The change in aggregation level is smooth as the user slides a control knob. The DTVC also provides a few predefined aggregation levels (week, month, quarter and year) with an animated transition between the current aggregation level and the predefined level. Animation is also used when selecting a bin / year to be in focus (discussed in the navigation section).

The use of animation is required to assist the user in tracking the information gain / loss as aggregation level changes. One of the main problems with existing solutions that provide several graphs in different discrete detail levels is the cognitive load when switching from one level to the other and identifying where is a period from one graph found in another. Animation is very useful in alleviating this cognitive load. Experimenting with the tool showed that it is highly helpful.

2.3.3 Visual cues on hidden / obscured information

Figure 1.b demonstrates the advantage of using the “trapeze” like visualization technique when the data exceeds the screen boundaries. The time labels in the bottom part remain anchored and the cyan time mark lines going out clearly indicate which time periods are in view and which periods are outside of the screen. This gives the user clear context and cues on where to look for information. The technique builds on some of the results shown in the City lights project [6]

2.3.4 Presenting bin information

The DTVC offers a wide range of information displayed on binned data. Some of the options (Figure 2) include: Highlighting the time period related with the bin under the mouse (A), the size of the highlighted triangle is proportional to the binned period’s length and its base location matches the period on the static time line. A tool tip with the exact period dates is also provided (C). The aggregated value of the data in the bin is displayed on the right-hand side in a vertical location matching this value (B). Standard deviation and minimum / maximum information for bin are drawn on the graph (D) as well as given in exact numbers (E). Finally the DTVC has its own menu for controlling the visualization and can therefore be independent of the hosting application. The information in A,B,C,E changes according to the bin under the mouse (when dragged with a right click). This supports **active exploration** of the time series by the user and prevents information clutter (only the desired information, under the current POI is displayed).

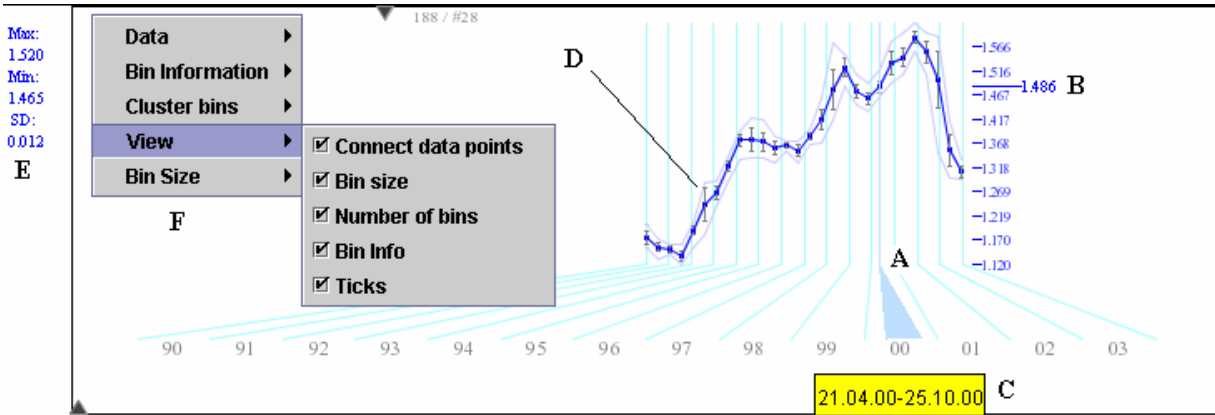


Figure 2 Visualizing binned data information. A rich menu is provided for dynamically controlling the desired visualization. Some of the visual cues are shown.

2.3.5 Navigation

Navigation is especially required when there is a need to expose information that is currently off the bounds of the screen. Several navigation options are introduced within the DTVC: Clicking on a year label (in the bottom static part) will center the time line part matching that year on the display, using smooth animation. It has already been demonstrated how off screen information can be easily identified using the trapeze line cues and thus it is always only one click away. The user can also use the arrow keys to scroll the time series back and forth. Finally clicking anywhere within the timeline will center the underlying bin. The last technique can be effectively used when first switching to a high aggregation level (so all the information fits the screen), identifying the desired period (using the bin information cues), clicking to center it in the display and then switching to a lower aggregation level. (much like the zoom/ pan trajectories described in [11])

2.3.6 Comparing two currencies

To support the task of comparing exchange rates of two currencies, the DTVC Supports the display of a second currency (overlay / separate vertical positioning) in the same aggregation level. Bin information will be displayed for both series and the spatial proximity between them facilitates the comparison. (Figure 3)

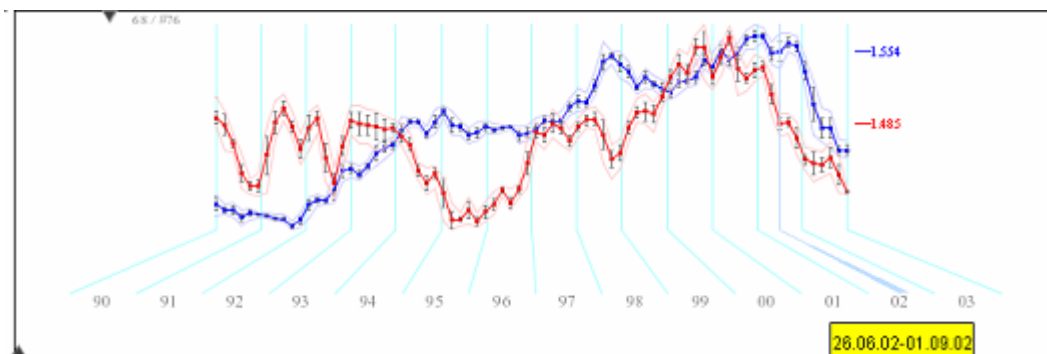


Figure 3. Two currencies are displayed in different colors. Bin information is displayed for both series as they are both presented on the same “trapeze”.

2.3.7 Marking

The DTVC supports the marking of desired time periods. The marks appear as colored boxes on top of the time series (Figure 4). The boxes are updated according to the aggregation level to cover the marked period, thus simplifying the detection of the same periods when aggregation level is changed. The box's vertical size is determined by the value span of the time series in this range, this assisting in detecting the behavior during this period.

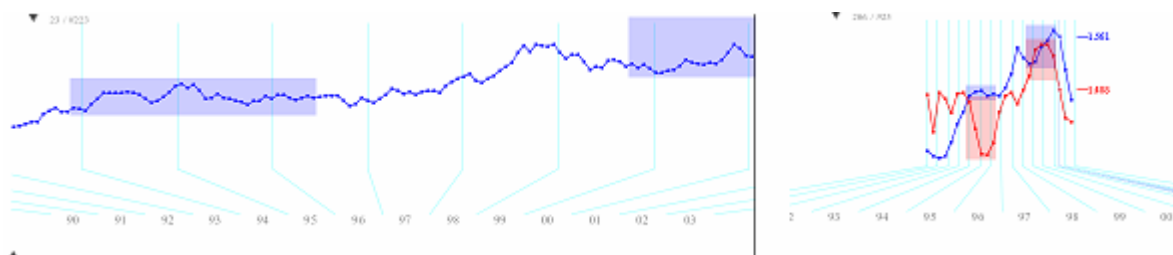


Figure 4. Marking time periods. The marks size is updated according to the aggregation level (left side shows a low aggregation level, while the right side has a high aggregation level and also displays the two series).

2.3.8 Color

When displaying two currencies together, color is used to identify each series. Since the visualization of each currency is composed of many components and details (such as the time line, bin information, axes, mark boxes etc.), it is highly important to maintain **color consistency** across all of these components. Still some color variations are required to differentiate the various components. The DTVC uses transparency and color manipulations to create different variations of series colors. This color information is also propagated to the hosting BinX application for use in other views. Finally, for maximum flexibility and accommodation of different users (e.g. color blind), the currency colors are configurable through central color selection menus.

2.4 The BinX application

The BinX application makes use of two DTVC components integrated with other views (Figure 5). A wide variety of **mix and match** visualization options are available, supporting various exploration techniques.

2.4.1 Detail and overview

The two DTVCs can be navigated independently so that each one can be focused on a different time period and in a different aggregation level. Another important navigation mode supported by BinX is linked navigation and marking across the two DTVCs.

When linked together (using a link button, Figure 5), each DTVC can still use its own aggregation level and information display. Yet, navigating in one DTVC will result in a matching animated navigation in the other one. The POI information (appearing on right-click drag) is also synced across the DTVCs. Marking a period with a mark box in one view will mark the matching period in the second. These linked views facilitate a detail + overview exploration of the data, when using one view with a high aggregation level (the overview) and one with a low or zero aggregation level (detail).

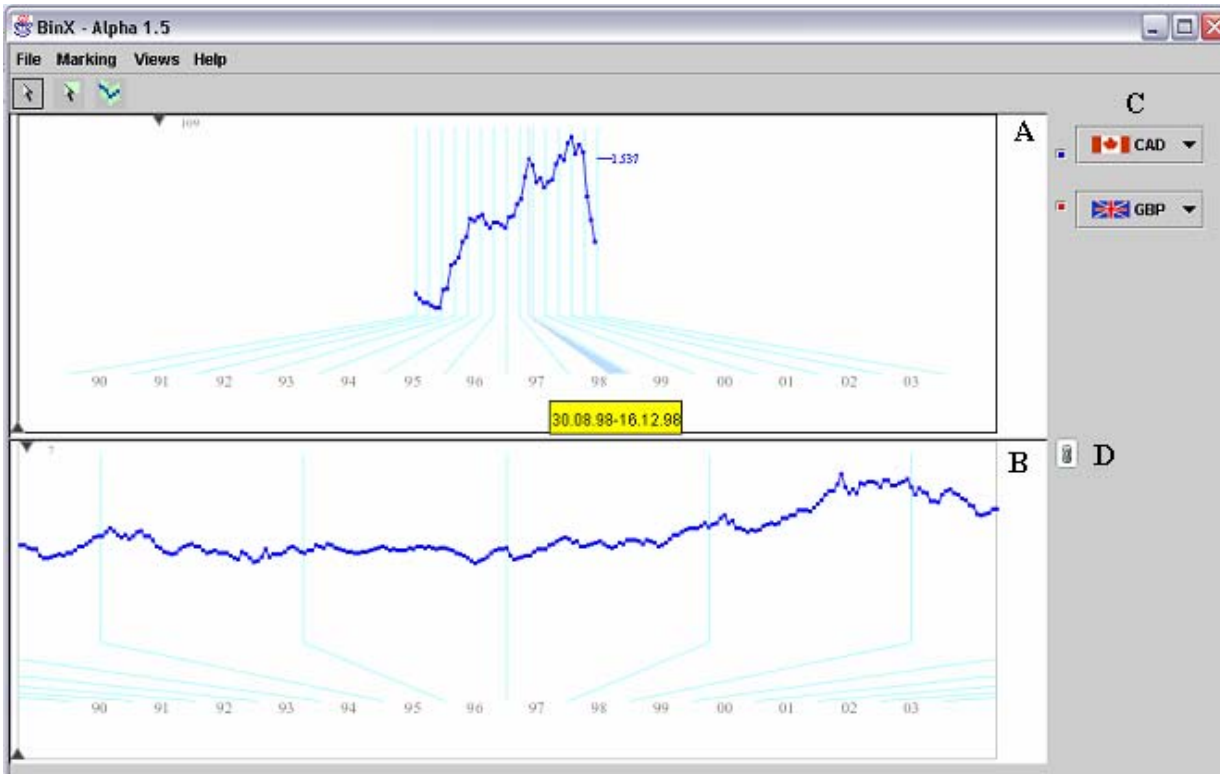


Figure 5. The BinX application. The displayed currencies can be selected (using C) as well as adjusting the associated colors. Two DTVC components (A,B) are displayed. Their navigation can be either linked or independent, using the link toggle button (D)

2.4.2 Clustering the time series

A fundamental analysis task is the identification of similar periods and patterns across the time series (these could be cyclic or acyclic). A useful method for assisting the user in such a task is the clustering of the binned data and reporting bins that share the same properties. This provides an initial mapping of bins for further exploration by the user. BinX offers a clustering mechanism that classifies the binned data and integrates this information into the visualization in a straight forward manner.. In section 3 a detailed use case of clustering is presented.

2.4.3 Scatter plot view and Brushing

When examining two time series (in this case two currencies), it is often useful to test for direct correlation between the two, ignoring the time dimension. BinX offers a scatter plot view, visualizing value pairs of the binned currencies and a computed regression line (Figure 6). This scatter plot is linked to the active DTVC view, so that changing the aggregation level in that view will update the scatter plot. Another interesting interaction is the ability to display and analyze only marked time periods in the DTVC (see the Marking section).

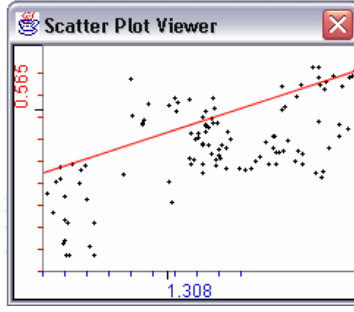


Figure 6. Each point $P = (C_{1t}, C_{2t})$, where C_{it} is the bin value of currency i in time t . A regression line is computed and displayed (in red).

2.5 High level implementation details

BinX was created using Java and has about 7000 lines of commented code, organized in several modules. The visualizations were implemented using the Java2D API. Most of the UI was implemented using Swing. The main modules are the DTVC component, the data processing module, the clustering services module and the visualization manager module (mastering all of the different visualization windows and taking care of synchronization).

The DTVC component was designed to be an independent component, so it can be simply integrated into any application. It can take care of its own UI and controls. It also offers a public API for external control and can generate events that an hosting application may use for synchronizing it with other views or controls (e.g. when being navigated). In section 2.4.1, it is demonstrated how two such components can be combined to create a powerful overview + detail model, using the API and events.

Clustering algorithms

The clustering algorithm uses the techniques describe in [3]. Initially, each bin is a distinct cluster. Then, iteratively, the two closest clusters (according to a distance function) are merged, forming a dendrogram with a single root cluster. The dendrogram can then be tipped off to get the number of clusters desired by the user. Three methods for computing the distance between clusters and the merged clusters are used: RMSD (Root mean square distance):

$$d_{rms} = \sqrt{\sum (b_{1i} - b_{2i})^2 / N} , \text{ where } b_{ji} \text{ is the value of the } i^{\text{th}} \text{ time interval in bin } j$$

When merging clusters their values are averaged: $b_i = (b_{1i} + b_{2i}) / 2$. This guarantees that the created clusters will have the same value range as the original time series. Another distance function that can be used is a normalized distance, which is good for clustering based on the shape of the underlying series.

$$d_{nm} = \sqrt{\sum (b_{1i} / b_{1max} - b_{2i} / b_{2max})^2 / N}$$

Using this distance is equivalent to first normalizing the binned data using the maximum value and then using d_{rms} . Merging clusters is done by averaging the normalized values. Finally the third distance function is the same as d_{nm} , only that the data is normalized using the maximum and minimum values $b_i = (b_i - b_{min}) / (b_{max} - b_{min})$ to create a better spread of the time series shape.

3. Scenarios of use

Following are three cases where the BinX visualization tool can be useful.

3.1 Finding matching time periods

A common task, when analyzing a currency exchange rate, is finding periods where the exchange rate level and its patterns (ascending, descending, complex shape or rate of change) match a given time period, such as the present week, month or quarter. This also applies to other financial indicators such as stock quotes, economic indexes etc.

The task is often performed when creating periodical financial reports or even when checking one's foreign currency investment portfolio on-line to check its state along a period of time. The author's bank offers such an on line service showing aggregated data, but making a full comparison of the underlying data is impossible, due to the lack of dynamic exploration.

Using BinX, one can start viewing the time series in a high level of aggregation or in an aggregation level matching the questioned period (the period's length can change based on the desired report, so having the ability to dynamically change the aggregation level is extremely handy). This high level view already gives a rough notion on what periods could resemble the questioned period. Applying clustering to the data, one can immediately get a view of matching periods (Figure 7a). A cluster viewer, (Figure 7b), exposes the characteristics of the clustered data. Several clustering methods can be used to match timeline values or shape.

By linking the second DTVC, showing a lower level of aggregation, one can "walk" on the clustered bins and immediately get a view of the actual data in the second DTVC.

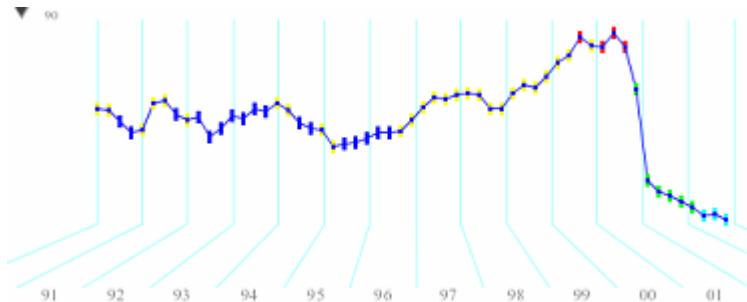


Figure 7a. Clustered time series. Each bin represents a quarter and is colored based on the cluster it belongs to (the number of clusters is user controlled). In this example there are 5 clusters colored yellow, blue, red, green and cyan)

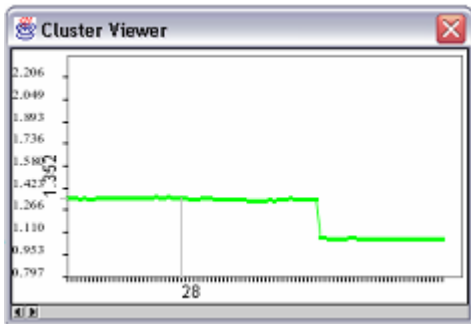


Figure 7b. The cluster viewer depicts the characteristic behavior of the cluster. The arrows on the bottom left allow navigation through the different clusters.

3.2 Detecting correlation between currencies

Finding the relation between two currencies or in general two time dependent variables is often a difficult task when viewing long time series. BinX allows a quick view of the two selected currencies in a single DTVC. Both currencies will be displayed using the same aggregation level and can be manipulated and navigated together. To avoid information clutter one can use the additional DTVC to depict other information (for instance DTVC 1 can show the two series, DTVC 2 can show only one series with full bin information or in a lower aggregation levels). The linked navigation mode will keep all of the information synchronized. (see Figure 8).

Another useful view that can be used is the scatter plot view. It enables one to readily get the direct correlation (and a regression line) between the two currencies in the current aggregation level. BinX also allows the user to mark periods with potentially high correlation and test only these periods in the scatter plot, reducing the amount of noise. The latter method is extremely useful when combined with a high level aggregation view, where periods with visually low correlation can be filtered out, using the marking method.

Finally, a user can keep one currency fixed as well as the visualization parameters (open views, aggregation levels, focus points and marked periods) and dynamically change the second currency from within the available currency set. This will be instantly reflected in all open views. Thus, one can quickly compare how different currencies and their features match a specific currency.

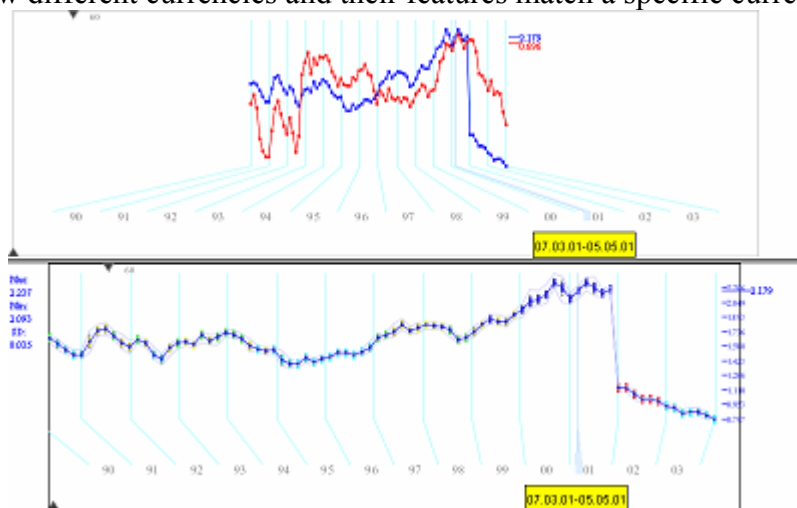


Figure 8. Viewing two currencies across two linked DTVCs. The bottom DTVC shows full bin information while the top one shows the two currencies without causing clutter.

3.3 Marking and tracking time periods

In time analysis tools there is often a need to select specific time periods and use their underlying data for further processing / viewing. This becomes a challenge when the time series is long and the desired time span cannot fit the screen in a single view. For instance the TimeSearcher tool [7] is largely based on the ability to mark time periods with query boxes. A major problem with this tool is that it does not scale well to long time series.

BinX offers an easy mechanism to mark time periods on long time series. In any aggregation level the user can mark the period of interest, creating a marking box. This box is properly adjusted when changing the aggregation level, thus making it easy to track a specific time period across detail levels.

Furthermore, marking a long period of time is possible by first switching to a higher level of aggregation, where the desired time span fully fits the screen. Another interesting option supported by BinX is using the two DTCV to display different parts of the time line that otherwise would not fit a single display and then marking the desired sections on each DTCV.

Using the DCTV API, the marked data can be extracted by the hosting application for different purposes (for instance the BinX uses this information to feed the scatter plot view).

4. Related work

Time series have been visualized for a long time, using conventional line chart or bar charts. However, most visualization techniques did not address the handling of long time series. Recently some work has been done to effectively visualize long and multivariate time series.

4.1 SolarPlot

SolarPlot [2] uses dynamic aggregation to display a univariate time series on the perimeter of a circular time line using bar-chart like spikes. Controlling the level of aggregation is accomplished by changing the radius length of the circle, so that as the circle becomes smaller a higher aggregation level is used. In this work several key characteristics of a successful aggregated view are presented, these are all met by the “trapeze” view used in BinX:

- Automatic aggregation of elements: When narrowing the trapeze, data is automatically aggregated to fit (or is conversely less aggregated when expanding the trapeze)
- Interactive control of the aggregation level
- Context maintenance across aggregation levels: The fixed time line in the bottom part serves as a clear anchor when changing the aggregation level
- Clear feedback on the level of aggregation: Several visual feedbacks are provided. The aggregation level knob and its labels, the horizontal distance between the time tick lines, the highlighted triangular area shown under the mouse (POI) has a proportional size to the binned period.

The SolarPlot visualization lacks sufficient information on the binned data. Apart from the value representing the bin, no other information is provided. This can result in misleading visualizations. BinX, on the other hand, provides various visual indications on the binned data (namely min/max, standard deviation and clustering) that mitigate the information loss. Furthermore, the use of two DTVCs allows one to smoothly and easily get a view of both the bin and the data inside. The SolarPlot offers an awkward pop-up display of binned data.

Most importantly, the SolarPlot display does not scale well to data sets, where low aggregation levels will have more data points than the perimeter of the maximum circle fitting on the screen. For instance, the data used in this work with 5000 points will require a circle with a diameter of about 1600 pixels to display the non-aggregated data, which is beyond the size of normal displays. BinX, on the other hand, can display a long time line across the boundaries of the screen, while providing simple cues and intuitive means to smoothly bring these into focus.

Finally, depicting more than one variable in the same aggregation level on a circular display is somewhat problematic, especially when using spatial encoding. Each variable has to be displayed on a

different radius, which breaks the radius = aggregation level principle. BinX uses the regular time line metaphor that can effectively display even more than two variables.

4.2 Cluster and Calendar based visualization

The cluster and calendar visualization in [3] demonstrates the power of clustering time series for visualizations. The time series data is divided into time units (for instance hourly based data is divided into days), then time units with similar characteristics are clustered together. The visualization uses these clusters to provide an overview of the time series on a calendar view. BinX takes advantage of clustering to create an enhanced overview of the data in a similar manner.

Instead of using a calendar view of the clusters, which steps away from the data, BinX displays the cluster information on top of the aggregated time series display. This gives a solid context to the cluster information and lets one immediately drill down to the underlying data. Furthermore, BinX allows the user to dynamically change the time unit used for clustering (being the aggregation level used for view), thus giving the user much more flexibility.

4.3 TimeSearcher

TimeSearcher [7] is a tool for placing visual queries on a set of time series using time boxes. The purpose of TimeSearcher is quite different from BinX, but they both offer visual exploration of time series. One of the major limitations of the TimeSearcher tool is its ability to handle long time series and to place time boxes on such series. BinX provides a set of tools that allow one to easily navigate, manipulate and mark time periods on long time series and can serve as a possible extension to the TimeSearcher tool.

4.4 Interactive Visualization of Serial Periodic Data

Recent works [8][9] have used spirals to visualize time series. Such visualizations are extremely useful for detecting cyclic patterns, especially with the ability to dynamically change the cycle length. The data in these displays, however, is usually aggregated in a fixed level, which may lead to misleading visualizations. A possible extension to these tools is the use of dynamic aggregation, demonstrated in BinX. Combining the control of aggregation and cycle length can empower the user in the analysis process. In section 5.2.3 a prototype system of BinX that tested such spiral visualizations is discussed.

5. Evaluation

BinX is still in a partial implementation stage, although most intended features are working. The dynamic aggregated view has proved to be very useful when used on the currency data sets. It enables an effective and smooth exploration of the data, focusing on the interesting periods (peaks, lows etc.). The clustering feature was only partially implemented and requires much more work. Following, is a discussion of the key strengths, weaknesses and some of the lessons learned

5.1 BinX Strengths

5.1.1 Overview + detail

Using the tool to explore the dataset showed that having two views, each with a different independent and controllable aggregation level, but with a linked navigation between them, is highly valuable for exposing the underlying patterns. This technique allows the user to effectively maintain context while exploring a long time series in different time periods and different detail levels.

5.1.2 Visual cues on binned data

The different visual cues provided on the binned data and the way they are integrated on the display help in preventing false visualizations, without cluttering the display. Bins with a lot of “noisy” data can easily be detected using the standard deviation or min/max cues. The clustering information showed on top of the graph greatly helps in comparing time periods. One of the interesting lessons learned when developing the system was the effectiveness of providing dynamic information on the current POI (under the mouse), which allows for active exploration of the data, while avoiding unwanted clutter.

5.1.3 Animation

The smooth transition between aggregation levels and focus points effectively alleviates the cognitive load and helps the user track information. Earlier versions that did not use animation required a relatively long adjustment period when switching to a new aggregation level. The animation is extremely useful when jumping directly to different aggregation levels (as opposed to using the aggregation knob) and when changing the focus period (time shift).

5.1.4 Mix and match

BinX offers flexible ways to explore the data and thus can be adapted to accommodate the desired analysis technique. Each DCTV can be configured to display different currencies and different bin information, it also has its own aggregation level and focus point. At the same time, these properties or some of them can be linked across the two DCTVs. These abilities have proven to be highly useful, especially when there is a need to adapt the display model to the data explored. Many times the nature of the data (in this case specific currencies) requires different visualizations and the mix and match approach supports that.

5.2 weaknesses and problems

5.2.1 Handling larger scale data sets

Due to time limits and lack of data the tool has not been tested with much larger data sets (e.g. 50,000 data points). Although the visualization principles will probably still hold for such a number of points, the tool has not been fully optimized and can therefore have performance problems. For instance, although not drawing any off screen information, the tool still makes several computations for non-displayed data. The clustering algorithm also needs optimization, as it was implemented only in a “proof of concept” level.

5.2.2 Adaptation to smaller size displays

In the current implementation, the DCTV is not adapted to work on a small display area. It may be desirable to use the suggested techniques on small display sizes that cannot fit even shorter time series. In such case, further work needs to be done.

5.2.3 Spiral view

In early stages of the development, a spiral view of the time series was tested (Figure 9). This view used spikes to represent the data points (much like the SolarPlot described in section 4) and allowed one to change the size of the spiral, thus changing the aggregation level. However the prototype version demonstrated the main problem with this view. There is not enough room to effectively display more than one variable, while still allowing reasonable tracking of temporal changes (especially in low aggregation levels, where the spiral size is too small). Also, when the display space is insufficient for all data points, there is no natural visual extension to the data (as opposed to the “trapeze” view and its off screen cues). At the same time this model is of virtue and the Future work section discusses some possible directions.

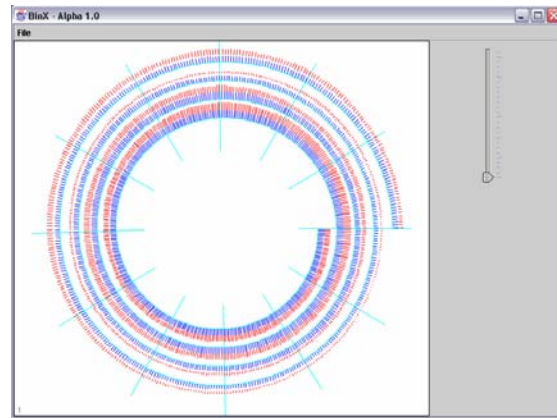


Figure 9. A spiral view of the data was tested in an early prototype.

5.2.4 The Scatter plot view

Depicting the correlation between two currencies in different aggregation levels, did not always provide useful insight. This can very well be the nature of the exchange rate data that is very smooth by nature. In such a case, changing the aggregation level does not change the correlation much. It is highly possible though, that with much noisier data, using higher levels of aggregation can filter some of the noise and expose the underlying correlation.

6. Future work

Several improvements can be made to BinX and the DCTV component to make them much more useful and powerful.

The most interesting extension to BinX is integrating it with the TimeSearcher tool querying functionality. Such integration can build on the flexible navigation, multi-level view and time period marking functionality offered by BinX to pose visual queries on long time series. This task is not supported by the TimeSearcher tool, which has limited navigation. The queries themselves can be carried out either on the non-aggregated data or on the binned data (assisting in mitigating noise effects). One other possible extension is to augment the query mechanism to support queries on the bin properties (for instance selecting only time series whose marked bins follow a specific distribution, etc.)

Another possible extension is allowing a flexible number of DCTV views. As mentioned in the previous section the mix and match abilities of BinX were very useful in the analysis tasks, either for creating overview + detail or for focusing on different time frames. Offering the user the ability to

create as many views as needed for the tasks, with richer methods of linking and placing could facilitate better analysis.

Despite the problems detected with spiral aggregated visualizations, they can still be useful in detecting cyclic patterns. Further work can explore how to better visualize information on a spiral while allowing control of both the cycle length and the aggregation level. One possible solution is using color intensity to encode the values instead of spikes (see [9]). Another option is using multiple linked views on the data, one with a spiral view and one with the “trapeze” view.

Finally, the clustering mechanism can be made much more flexible. Instead of clustering time intervals based on the non aggregated data they contain, it is possible to cluster the time intervals based on any lower level aggregation data contained in them. Another possible extension is letting the clustering algorithm work only on selected time intervals (using the marking mechanism), thus allowing for finer tuning of the clustering.

7. Conclusion

I have introduced BinX, a visualization tool for dynamic exploration of long time series. The tool uses key information visualization techniques to support effective analysis of the data. Most prominent are its support for smooth navigation and control of aggregation level. Experimenting with the tool revealed the strength of its mix and match capabilities. Two multiple view windows can be used in a variety of ways to accommodate the needs of the user and the nature of the data. BinX is useful as a standalone tool, yet its visualization component was designed, so it can be independently used in any hosting application, offering data exploration services and being integrated with other views.

Bibliography

[1] M. Wolfgang , H. Schumann , *Visualization Methods for Time-Dependent Data – An Overview (WSC 2003)*

[2] Mei C. Chuah and Roth, Steven F. (1998) "Dynamic Aggregation with Circular Visual Designs" *Proceedings of Information Visualization, IEEE, North Carolina, October 1998.*

[3] J. J. van Wijk and E. R. van Selow. Cluster and calendar-based visualization of time series data. In *Proc. IEEE Symposium on Information Visualization*, pages 4-9, Oct. 25-26 1999.

[4] FXHistory: historical currency exchange rates
<http://www.oanda.com/convert/fxhistory>

[5] Shneiderman, B., “The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations”, *Proc. of IEEE Symposium on Visual Languages*, Los Alamos, Sept. 1996, pp. 336-343.

[6] Zellweger, P. T.; Mackinlay, J. D.; Good, L.; Stefik, M. J.; Baudisch, P. City Lights: contextual views in minimal space. *ACM Conference on Human Factors in Computing Systems (CHI 2003)*, Extended Abstracts; 2003 April 5-10; Fort Lauderdale; FL. NY: ACM; 2003; 838-839.

- [7] Hochheiser, H. Shneiderman, B. Visual Queries for Finding Patterns in Time Series Data University of Maryland, Computer Science Dept. Tech Report #CS-TR-4365, UMIACS-TR-2002-45
- [8] J.-V. Carlis and J.-A. Konston. Interactive visualization of serial periodic data. In *Proc. of the ACM Conference on User Interface Software and Technology (UIST)*, San Francisco, CA, 1998.
- [9] M. Weber, M. Alexa and W. Mueller, Visualizing Time-Series on Spirals, in: Proc. of the IEEE InfoVis Symposium on Information Visualization (IEEE Press, Los Alamitos, CA, 2001) 7-14.
- [10] L. Tang ; B. Shneiderman Dynamic Aggregation to Support Pattern Discovery: A case study with web logs CS-TR-4345
- [11] Furnas, G. W., and Bederson, B. B.: *Space-scale diagrams: Understanding multiscale interfaces*. In Proceedings, CHI95, Human Factors in Computing Systems, 1995.