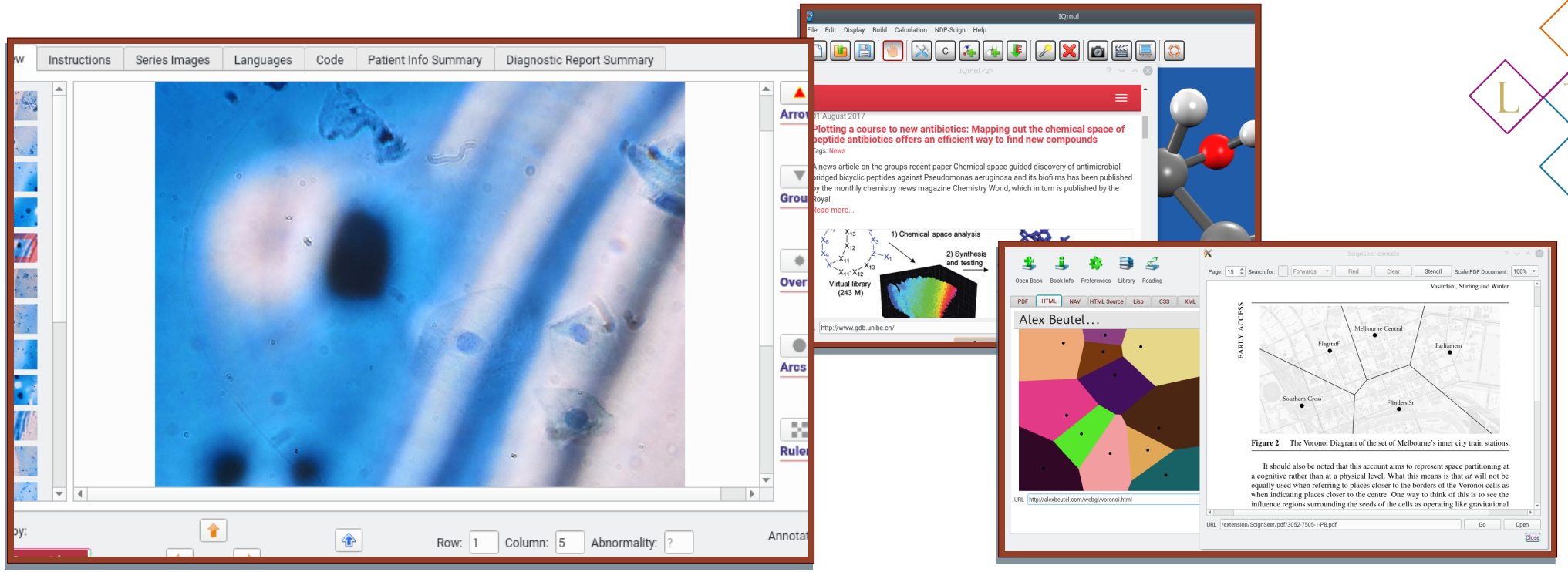
Dataset Creator

("dsC")



Linguistic Technology Systems (LTS)
Amy Neustein, Ph.D., Founder and CEO
amy.neustein@verizon.net
(917) 817-2184

Linguistic Technology Systems

Group 1: Features of Dataset Applications

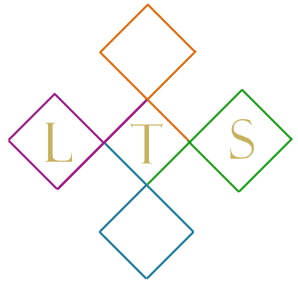
User Interface Features Typical of Dataset Applications

The code for each dsC data set includes a customized "Dataset Application" which displays individual samples and groups of samples via 2D, 3D, and native-compiled GUI controls. Each Dataset Application can thereby make use of advanced visual and interactive features that are uniquely possible when using customized, native-compiled GUI classes. The following screenshots will show several examples of these features, including:

Specialized Top-Level Controls Tree Widgets, Stacked Widgets, and Graphics Scenes.

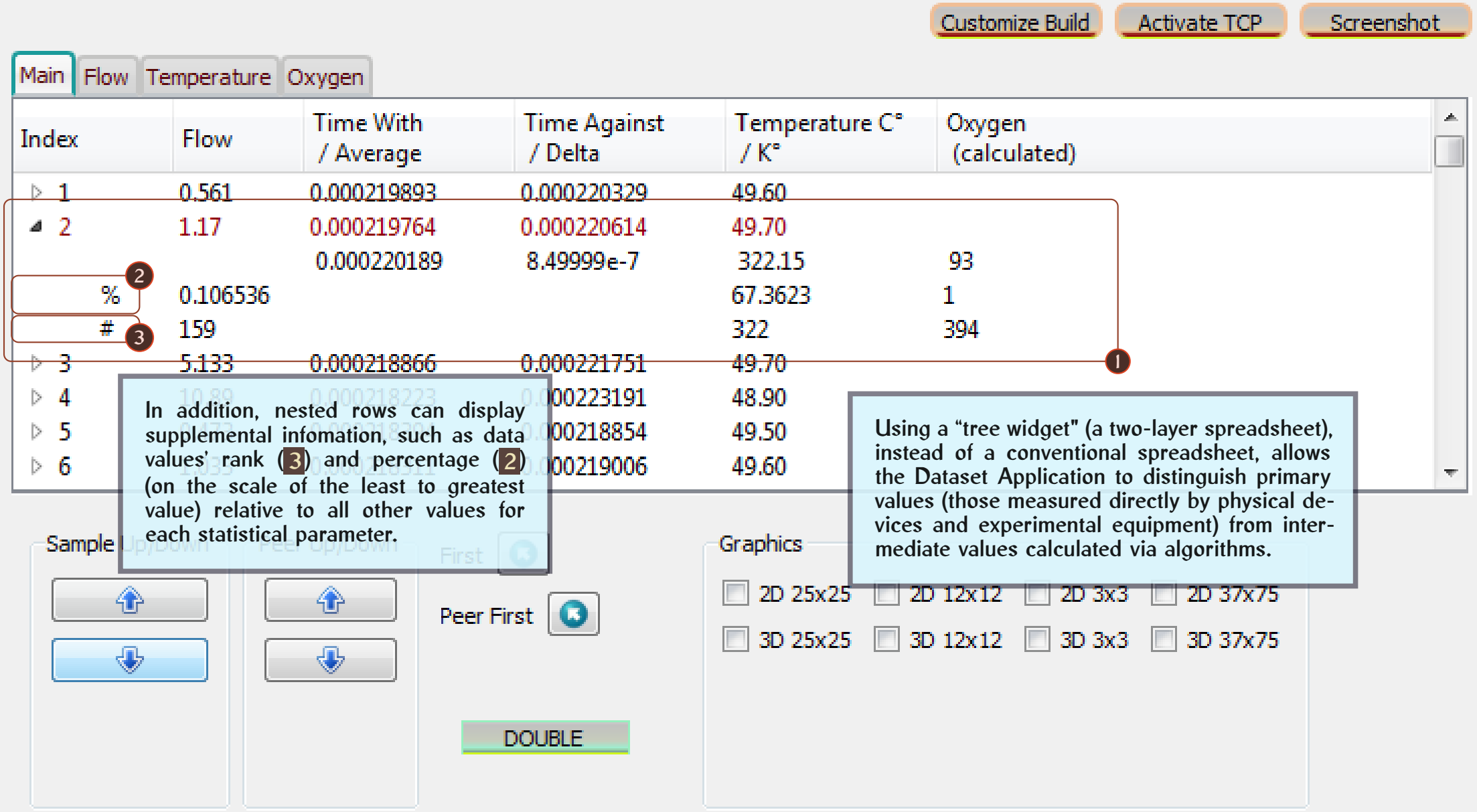
Context Menus Systematically organize functionality around UI layouts.

Multi-Window Displays Divide application functionality in multiple specialized top-level windows and/or dialog boxes.

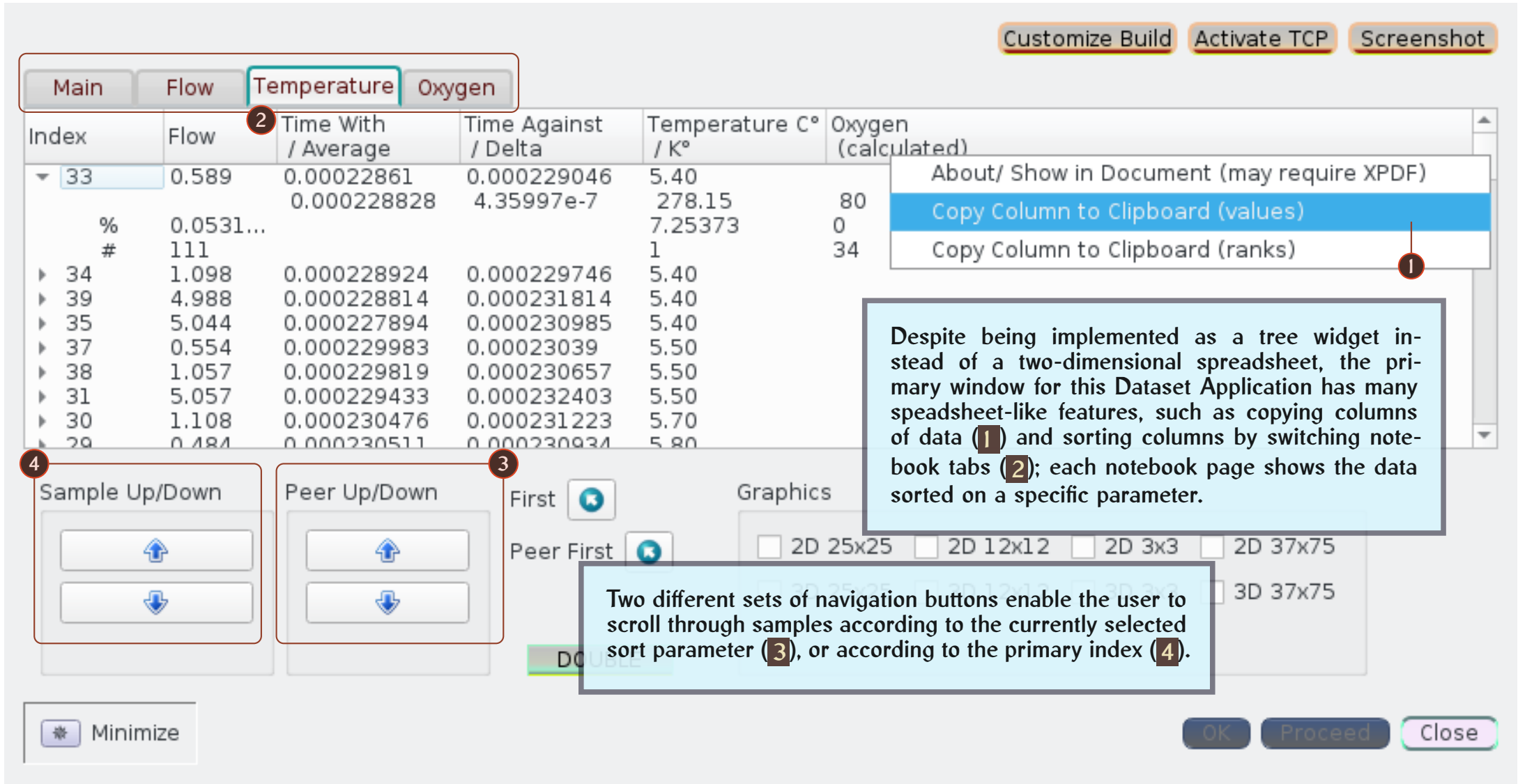


Linguistic Technology Systems

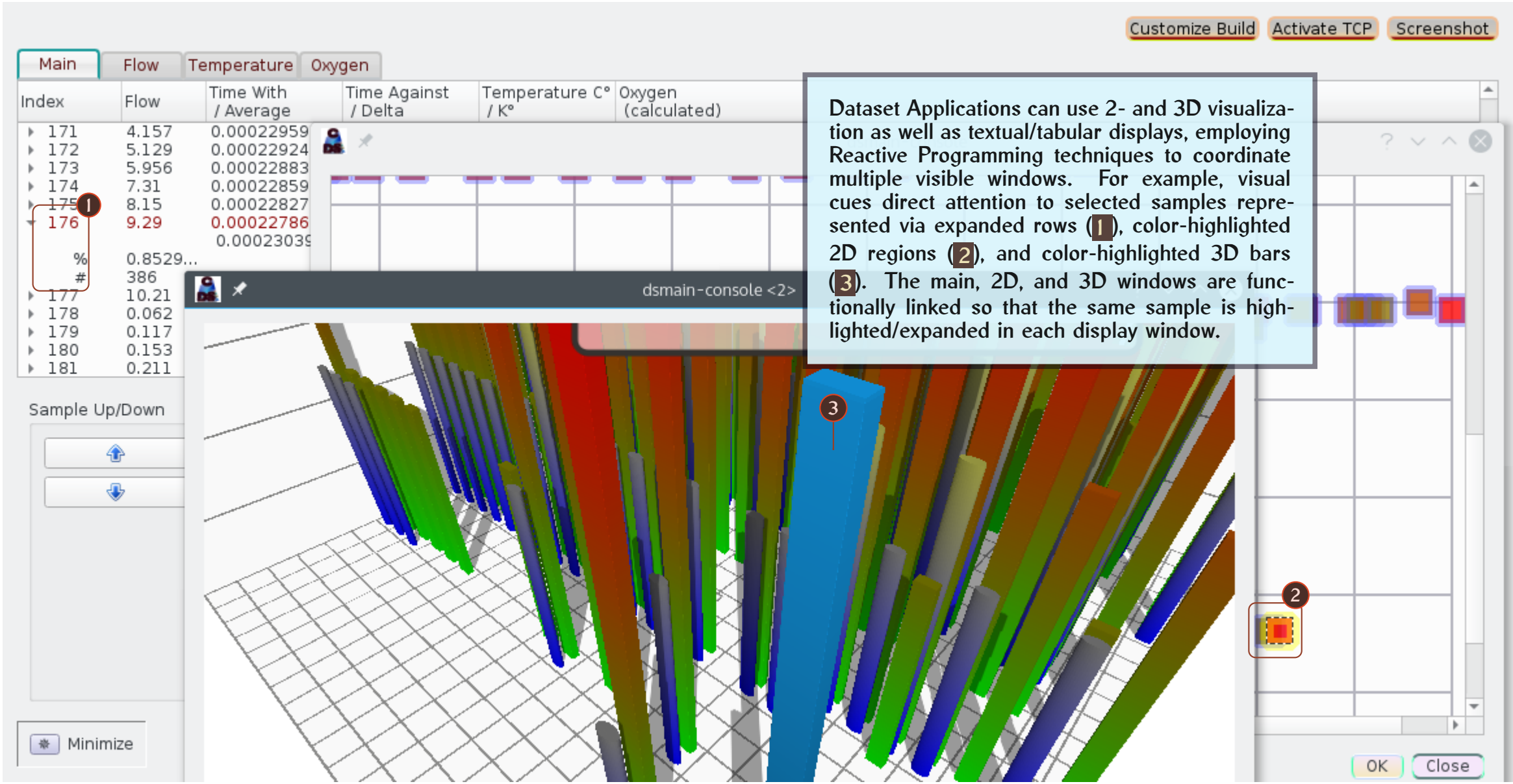
Initial Application Window



Interacting with the Main Window

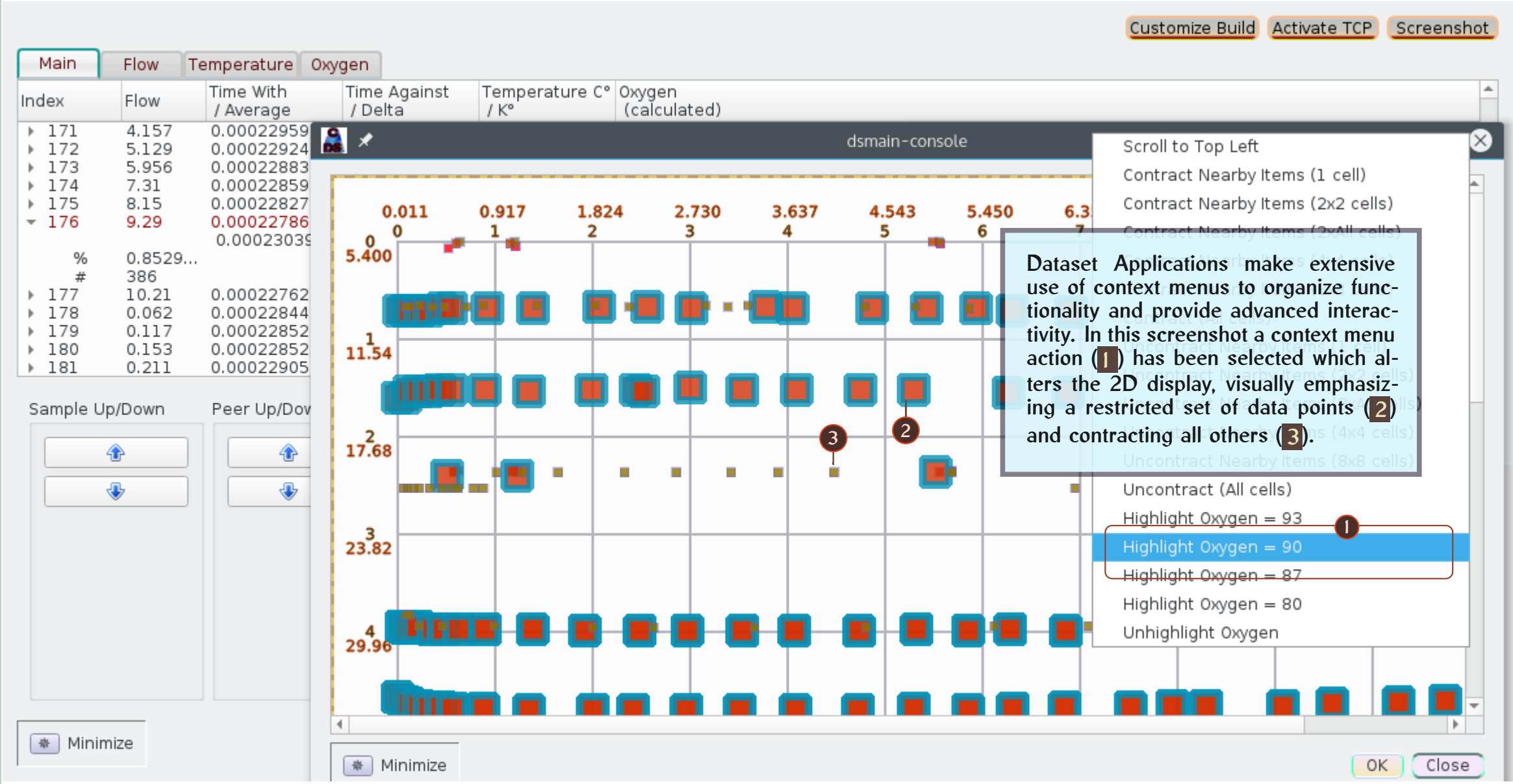


Coordinated Data Visualization



Dataset Applications can use 2- and 3D visualization as well as textual/tabular displays, employing Reactive Programming techniques to coordinate multiple visible windows. For example, visual cues direct attention to selected samples represented via expanded rows (1), color-highlighted 2D regions (2), and color-highlighted 3D bars (3). The main, 2D, and 3D windows are functionally linked so that the same sample is highlighted/expanded in each display window.

Interacting with the Visuals



Dataset Applications make extensive use of context menus to organize functionality and provide advanced interactivity. In this screenshot a context menu action (1) has been selected which alters the 2D display, visually emphasizing a restricted set of data points (2) and contracting all others (3).

Getting Information About Modeling Parameters

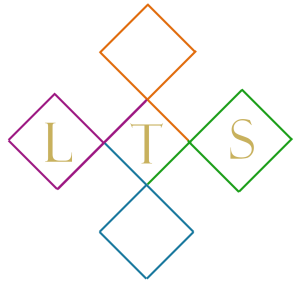
Using Dataset Applications as Pedagogical Tools

In addition to interactive visualization, Dataset Applications are useful tools for understanding experimental protocols and research methods. Within Dataset Applications, modeling units such as statistical parameters and record fields are visible in situ within a GUI — identified by labels, buttons, and other interactive micro-controls. As a result, users encounter modeling elements in a structured visual-intractive context. To learn more about modeling elements, Dataset Applications are equipped with several pedagogical features shown on the following screenshots:

"About" Dialogs Brief summaries of research trms and parameters.

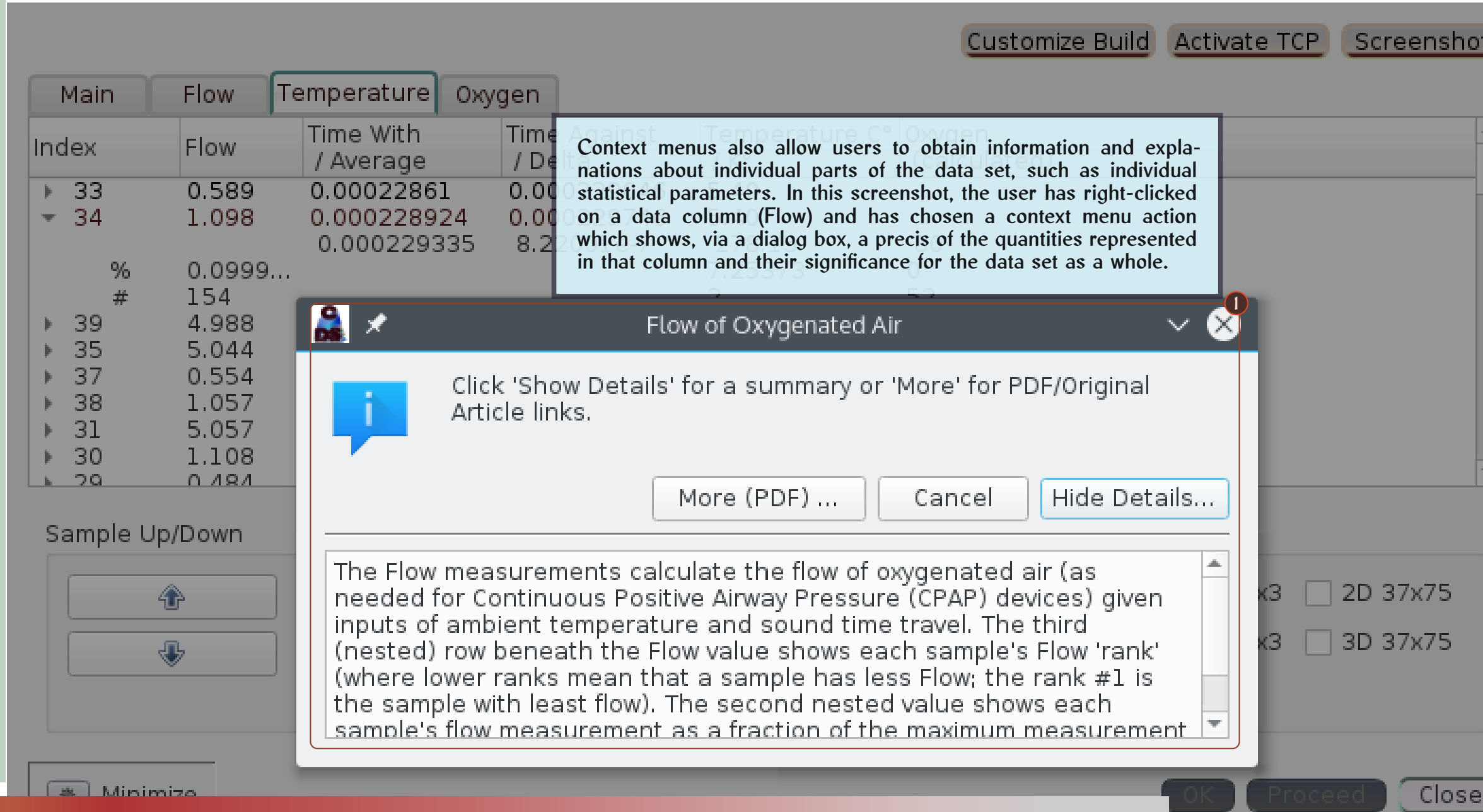
XPDF Links Links back to research articles read in an embedded PDF viewer.

XPDF Enhancements The XPDF viewer can be customized for each data set and included with dataset code, with extra features to integrate article or book texts with Dataset Applications.



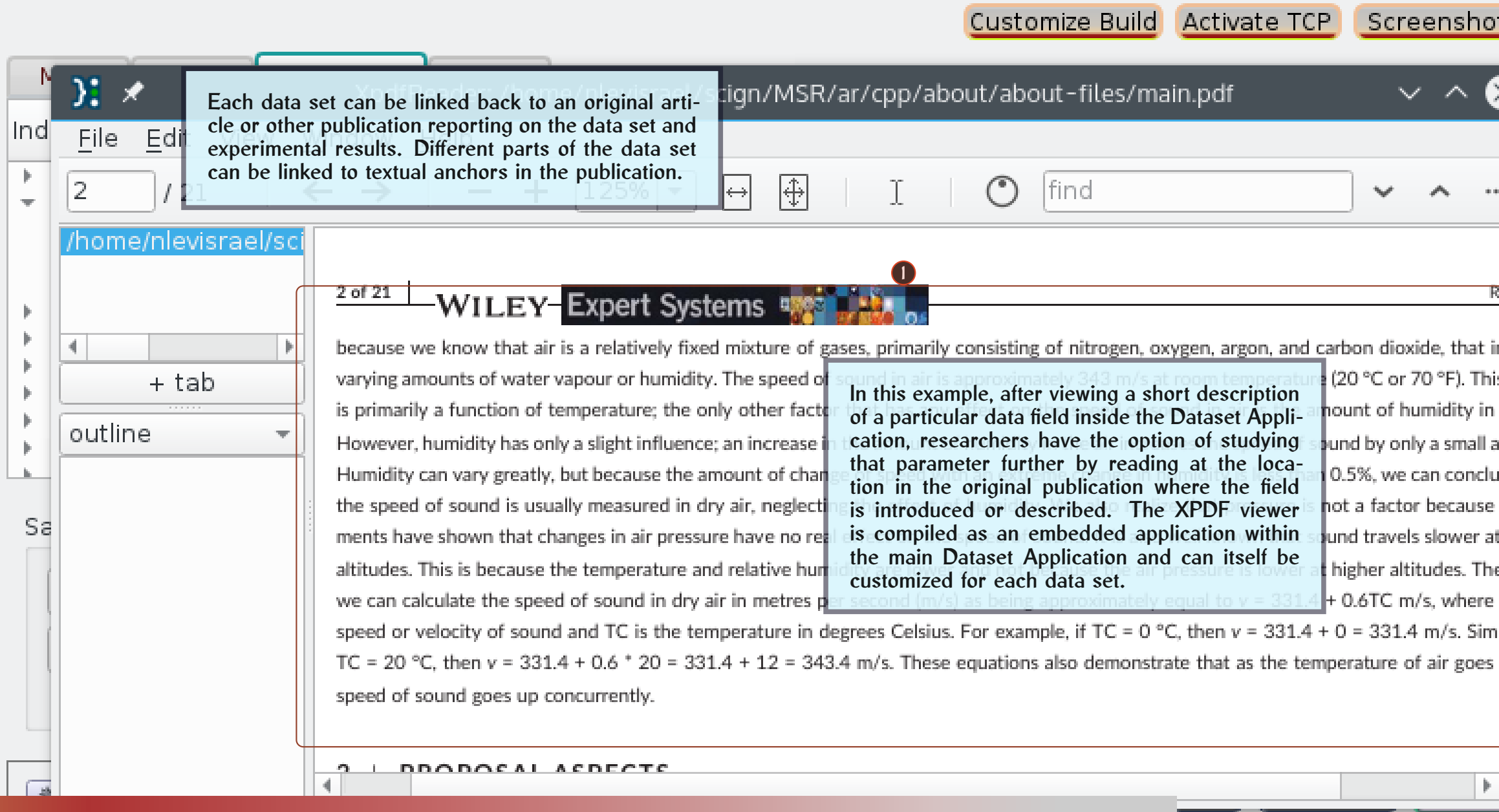
Linguistic Technology Systems

Obtaining Information About Parameters



Context menus also allow users to obtain information and explanations about individual parts of the data set, such as individual statistical parameters. In this screenshot, the user has right-clicked on a data column (Flow) and has chosen a context menu action which shows, via a dialog box, a precis of the quantities represented in that column and their significance for the data set as a whole.

Embedding XPDF



Testing and Fine-Tuning Dataset Applicationss

Tools for Editors and Developers

Although ordinary users can explore and visualize dsC data sets "Out of the Box", advanced users have many options for customizing their build of the application in terms of their specific roles and available 3rd-party libraries. These fine-tuning possibilities include:

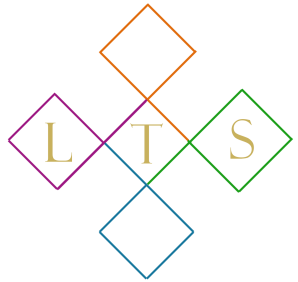
Test Suites Tools for creating and/or running test suites to ensure that the Dataset Application works across platforms.

Data Export Tools for reusing data in other projects.

External Libraries Some features like XPDF and 3D graphics require libraries that cannot be published with the data set in source code form. Advanced users can slect which of these libraries to incoporate into thir version of the Dataset Application.

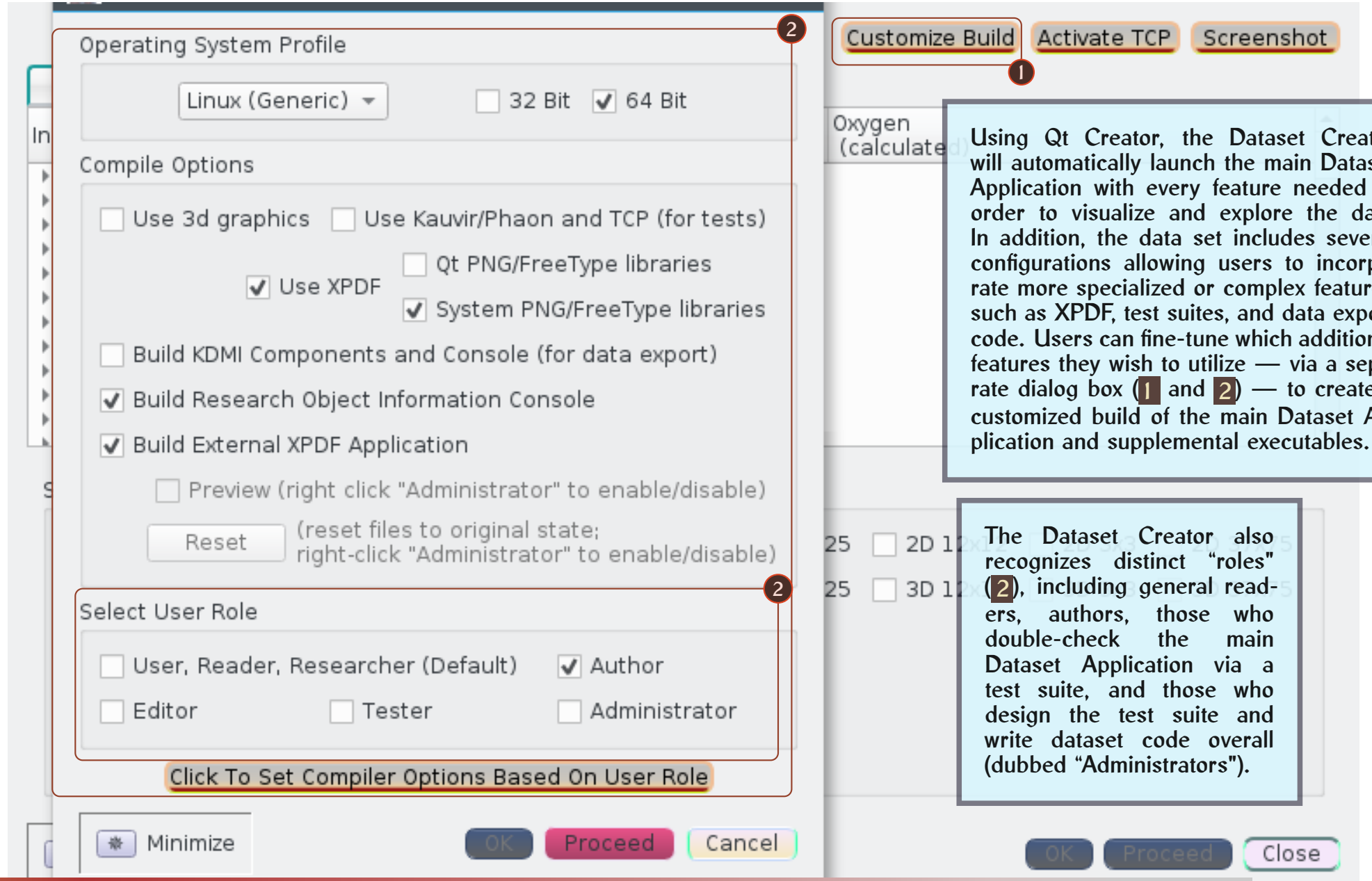
Scripting Data sets can comoile their own scripting environment to automate testing and manipulation of research data.

Networking Dataset Applications can use an embedded TCP server to communicate with other applications, enabling multi-application workflows (this is also how testing is implemented).



Linguistic Technology Systems

Configuring the Data Set Application

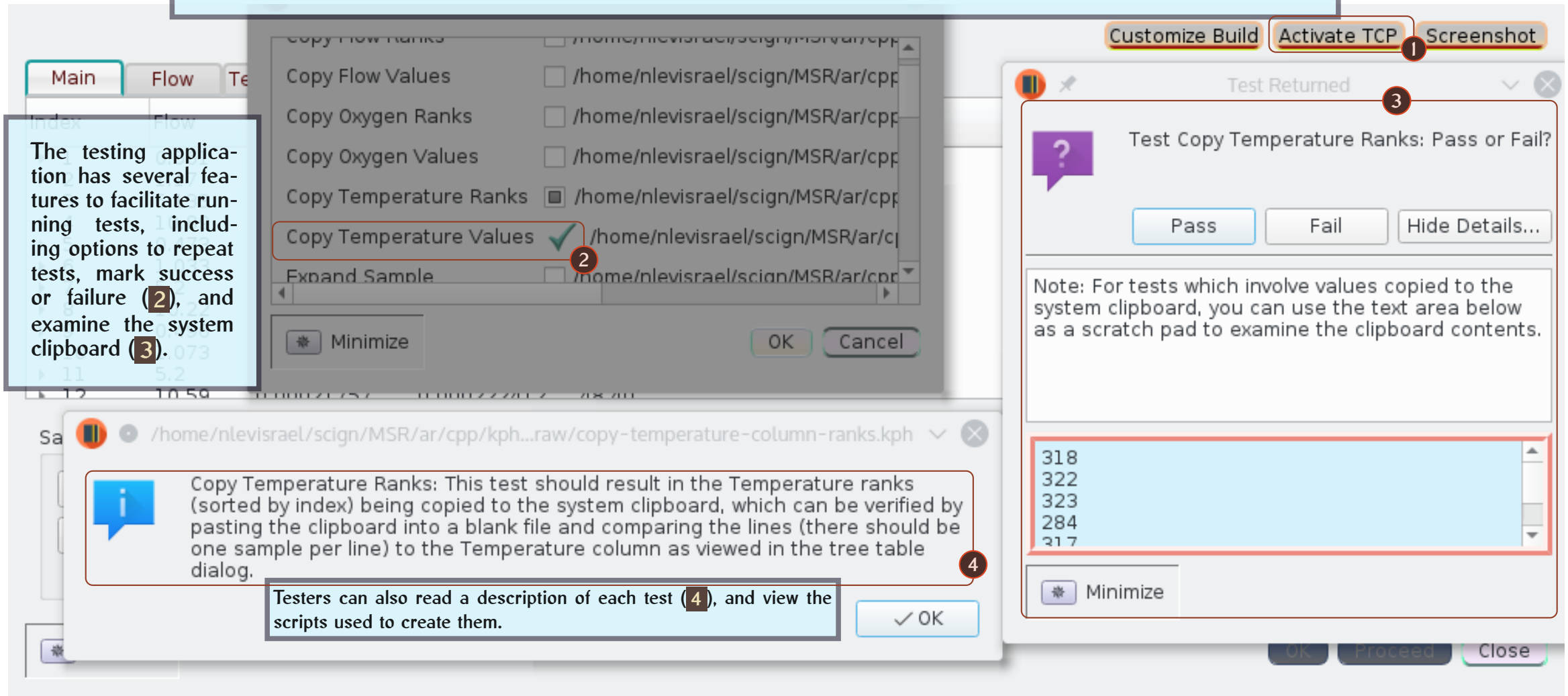


Using Qt Creator, the Dataset Creator will automatically launch the main Dataset Application with every feature needed in order to visualize and explore the data. In addition, the data set includes several configurations allowing users to incorporate more specialized or complex features, such as XPDF, test suites, and data export code. Users can fine-tune which additional features they wish to utilize — via a separate dialog box (1 and 2) — to create a customized build of the main Dataset Application and supplemental executables.

The Dataset Creator also recognizes distinct "roles" (2), including general readers, authors, those who double-check the main Dataset Application via a test suite, and those who design the test suite and write dataset code overall (dubbed "Administrators").

Testing the Data Set Application

Dataset Creator includes a sophisticated framework for building and running test suites to ensure that raw data is processed correctly and that User Interface components work properly on different Operating System platforms. This includes a separate testing application that sends instructions to the main Dataset Application via TCP (1).



The testing application has several features to facilitate running tests, including options to repeat tests, mark success or failure (2), and examine the system clipboard (3).

Testers can also read a description of each test (4), and view the scripts used to create them.

Features of Dataset Applications for Books

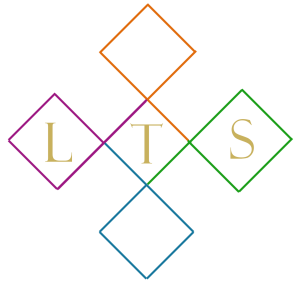
Datasets Ccompiled From Book Examples

The remaining screenshots demonstrate how data sets can be used even outside of a lab contxt generating experiment data. The pictured data set represents a corpus of linguistic examples mined from Wiley's *Blackwell Handbook of Pragmatics*. Creating data sets from book-length publications can encompass several steps:

Text Mining In the case of linguistics, this involves locating example sentences within linguistics texts and storing them as an independent corpus.

Canonical Formatting If possible, linguistics texts should be annotated so that extracting exmples can be automated.

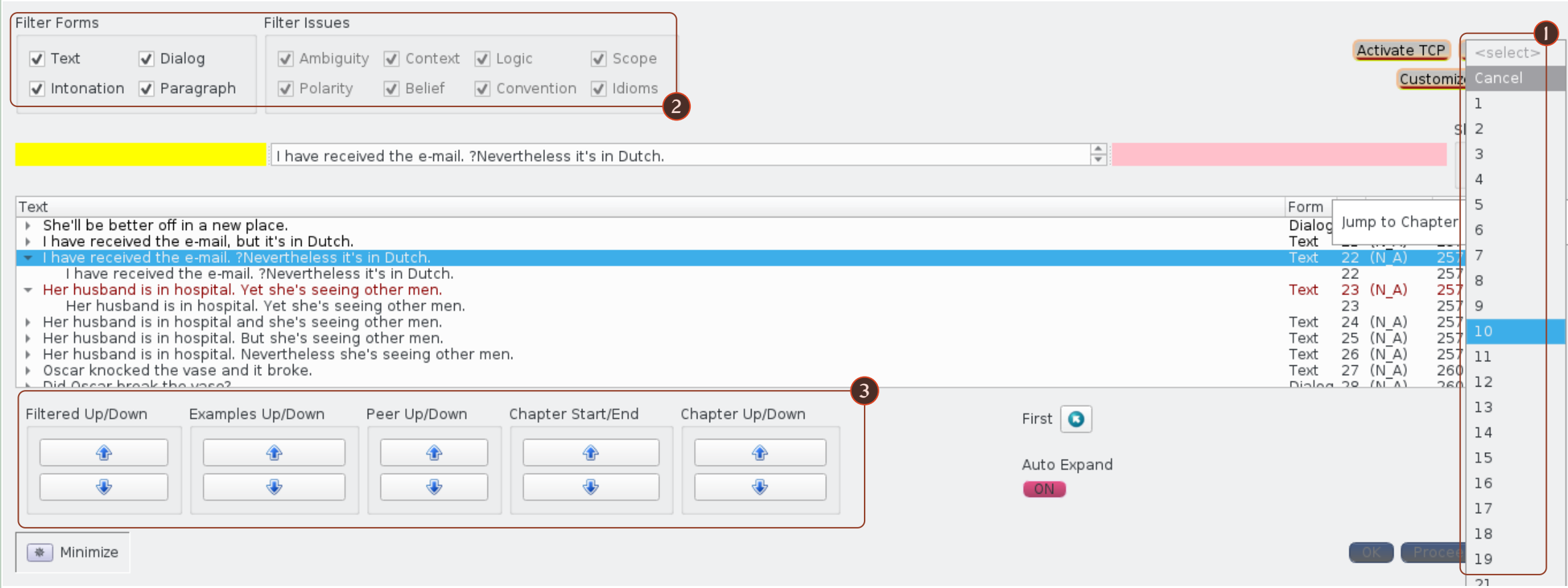
Annotation Linguistic corpuses are often annotated to identify structural details, beyond raw text, in each sample.



Linguistic Technology Systems

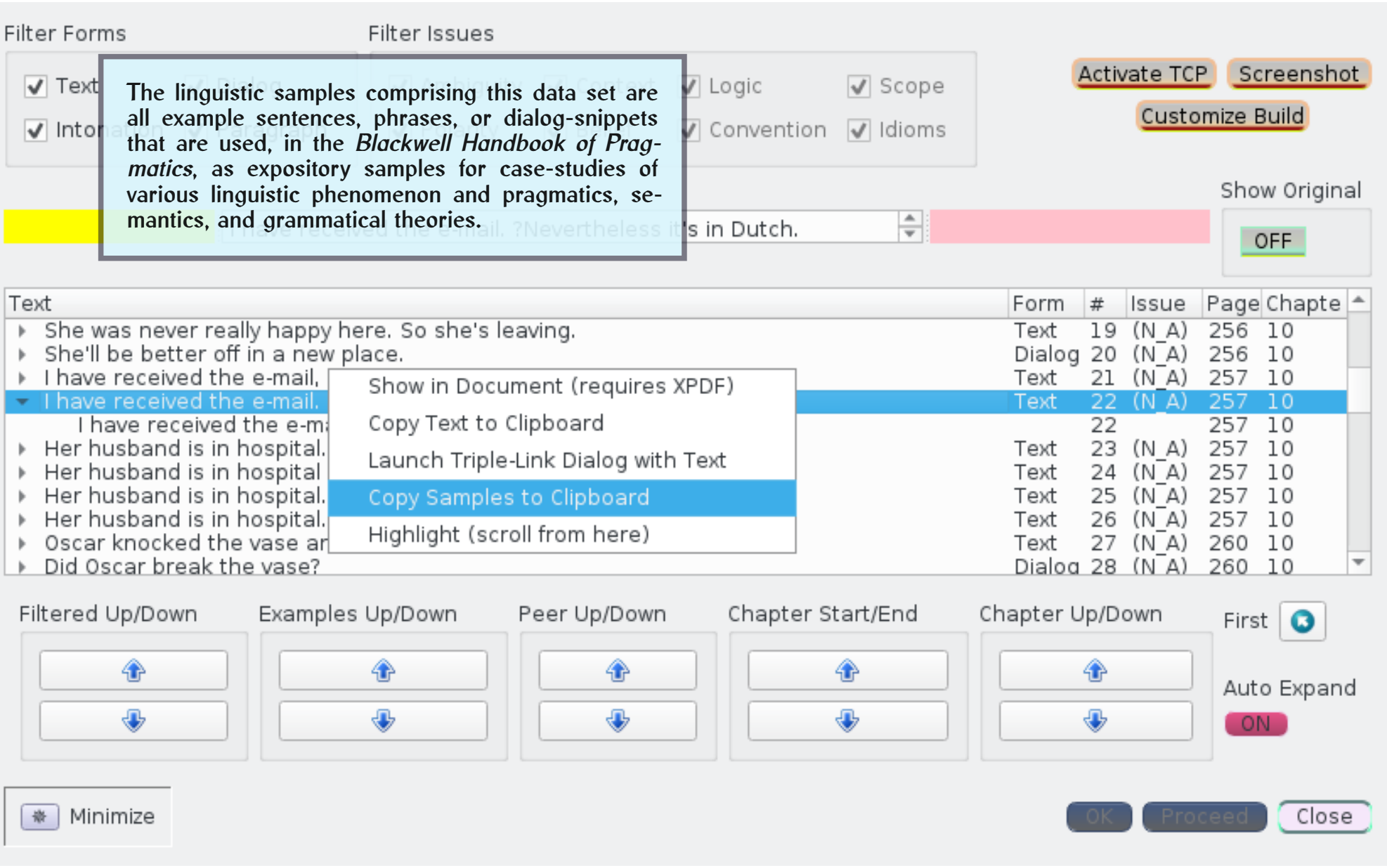
Creating a Data Set from a Book

This screenshot shows a linguistics dataset that illustrates several advanced interactive features made possible by the Dataset Creator's Qt-based front-end technology. Useful features include context menus embedded with drop-down selections (1) and button/checkbox groups for filtered scrolling through a list of samples (2 and 3).



Interacting with Data Samples

The linguistic samples comprising this data set are all example sentences, phrases, or dialog-snippets that are used, in the *Blackwell Handbook of Pragmatics*, as expository samples for case-studies of various linguistic phenomenon and pragmatics, semantics, and grammatical theories.



Linking Back to the Book

Filter Forms

Text Dialog

Intonation Paragraph

Filter Issues

Ambiguity Context

Polarity Belief

In France, Watergate wouldn't

Text

On the table.
Every bottle is empty.
She seized the knife and stabbed her husband.
The Boston Marathon will take place next week. Max thoug
My friends were under the impression that I was running a
Sue believes Luke has a child and that Luke's child will visit
In France, Watergate wouldn't have done Nixon any harm.
In France, Watergate wouldn't have done Nixon any har
In France, Watergate wouldn't have done Nixon any harm b
The crook paid them with fake money.
The crook thought he was paying them with fake money, b
We do not know much about this part of the brain, which p

Filtered Up/Down Examples Up/Down Peer Up/Down

Minimize

XpdfReader: /home/nlevisrael/scign/HP/ar/cpp/about/about-files/main.pdf

File Edit View Window Help

690 / 867 113% find

/home/nlevisrael/scign/HP/ar/cpp/ab

Outline

15. The Pragmatics o...
16. Pragmatics of La...
17. Constraints on Ell...
III Pragmatics and its Int...
18. Some Interaction...
19. Pragmatics and A...
20. Pragmatics and S...
21. Pragmatics and t...
22. Pragmatics and t...
23. Pragmatics and I...
24. Historical Pragma...
25. Pragmatics and L...
26. Pragmatics and C...
IV Pragmatics and Cogni...
27. Relevance Theory
28. Relevance Theory...
29. Pragmatics and C...
30. Pragmatic Aspect...
31. The Pragmatics o...
32. Abduction in Nat...
Bibliography
Index

After browsing through the data set, users can link back to the original text to see the current author's discussion of particular examples.

from the matched spaces to create a **blended mental space** with emergent structure. This creates a conceptual integration network of the form shown in figure 29.4. The generic space represents the structure shared by the inputs. The square in the blended space stands here for the emergent structure which arises in the blending.

So, for example, one way to understand the counterfactual in (6):

(6) In France, Watergate wouldn't have done Nixon any harm.

is to build a conceptual integration network that partially matches two input spaces with prominent aspects of the American political system and the French political system, respectively, and develops an emergent blended space

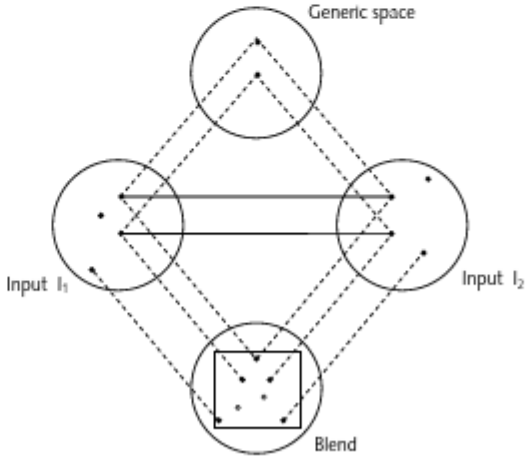


Figure 29.4 Diagram showing conceptual blending

A Linguistics Annotation System

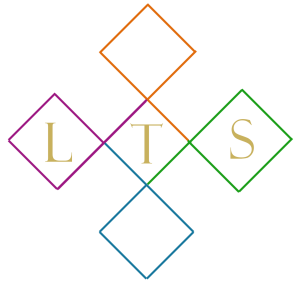
Tools to Facilitate Annotating Linguistic Corpora

The final three screenshots show an example of how a custom-signd application can facilotat the task of building an annotated corpus from a linguistics text. The components demonstrated here enable several strategies (which can be combined) for dscribing parsing structures and the logical composition of language samples:

S-Expressions Representing linguistic units as semantic and syntactic transformations triggered by words assigned to "functional" types.

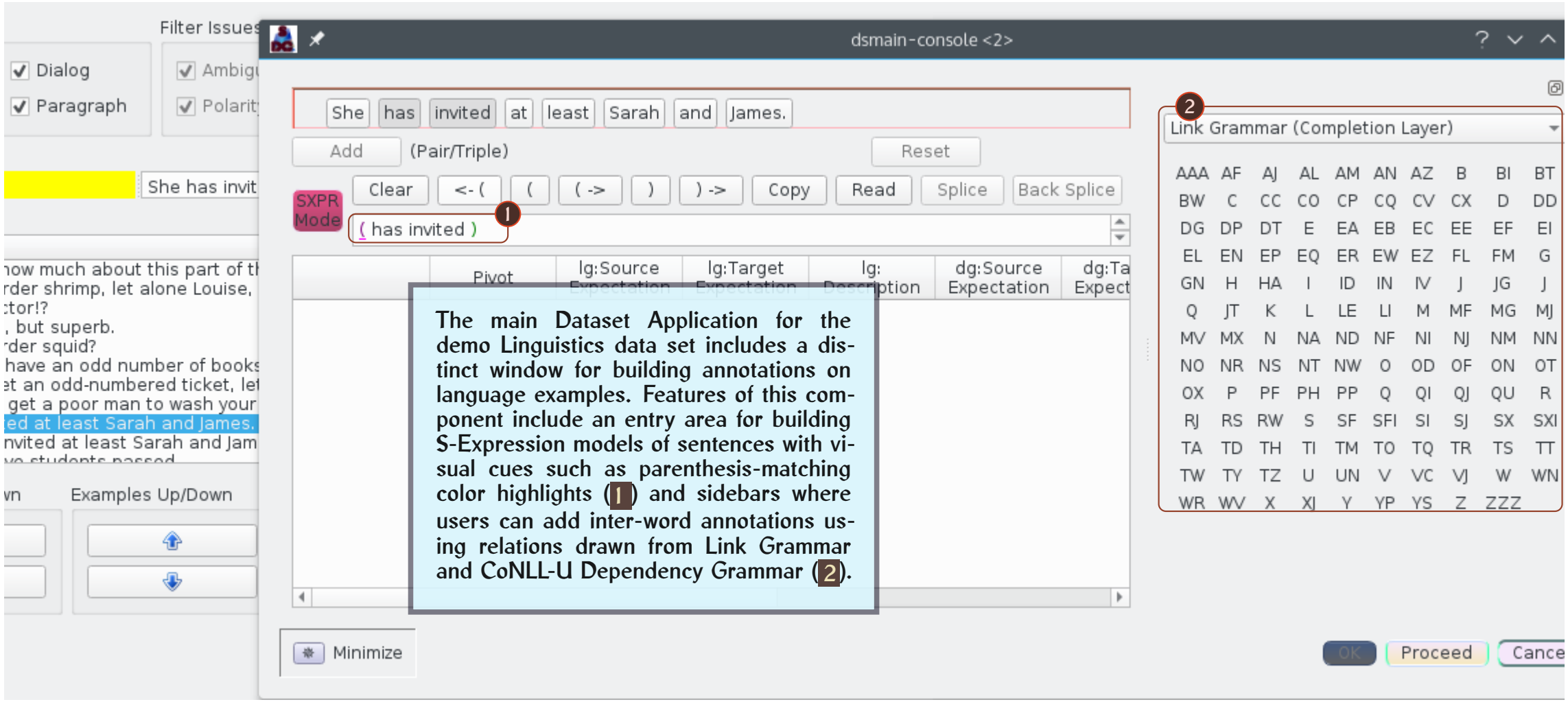
Deepndency Grammar Representing phrase structures viabinter-word syntactic relationships.

Link Grammar Representing linguistic structure via connectors internal to each word-sense. Inter-word links are activatd when each word in the pair has a connector compatible with the other word's connector. Intuitively, a connctor represents how one word's meaning or grammatic contribution can be "completed" by linking to a separate word.



Linguistic Technology Systems

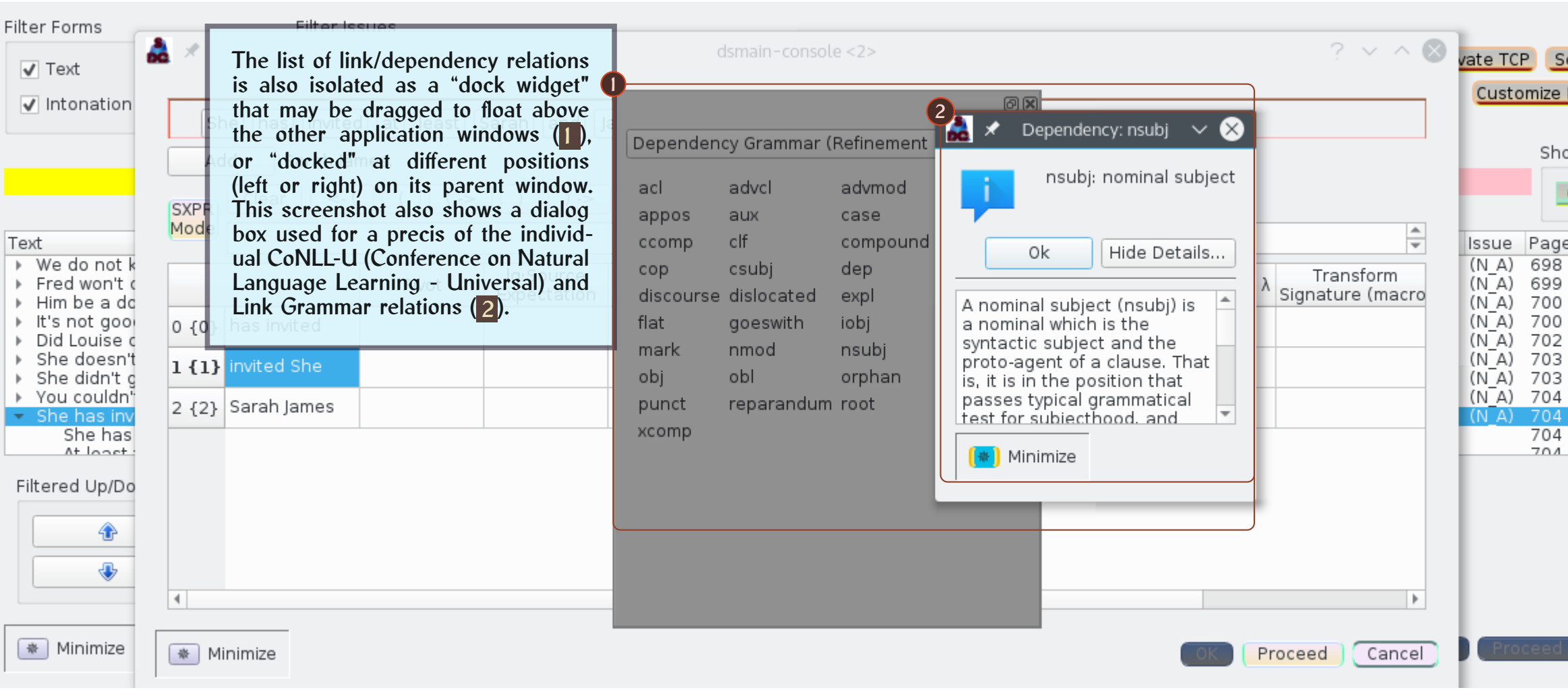
Building Parsing Models



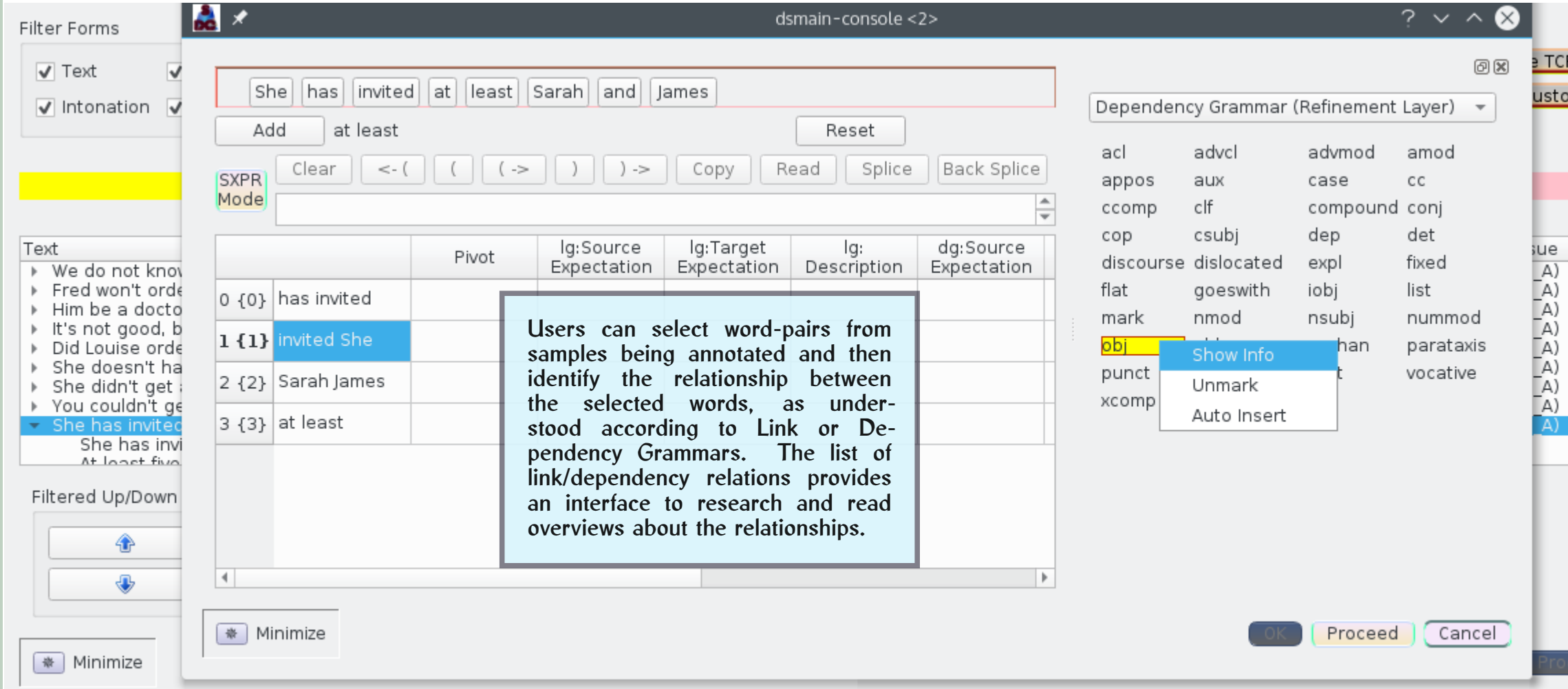
The main Dataset Application for the demo Linguistics data set includes a distinct window for building annotations on language examples. Features of this component include an entry area for building S-Expression models of sentences with visual cues such as parenthesis-matching color highlights (1) and sidebars where users can add inter-word annotations using relations drawn from Link Grammar and CoNLL-U Dependency Grammar (2).

Using Dock Widgets For Flexible Layout

The list of link/dependency relations is also isolated as a "dock widget" that may be dragged to float above the other application windows (1), or "docked" at different positions (left or right) on its parent window. This screenshot also shows a dialog box used for a precis of the individual CoNLL-U (Conference on Natural Language Learning - Universal) and Link Grammar relations (2).

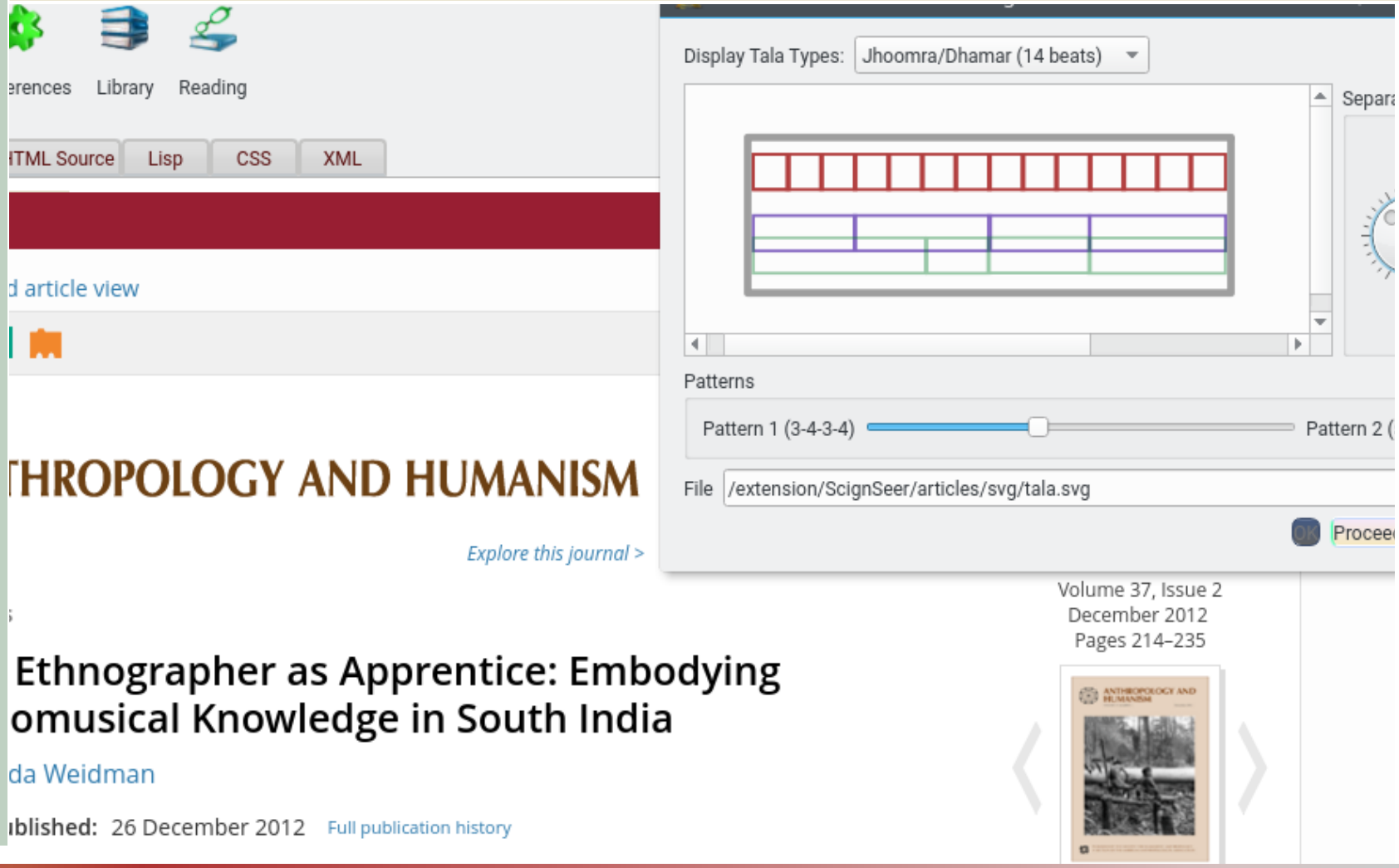


Link and Dependency Grammar Annotations



A3R Document Viewers

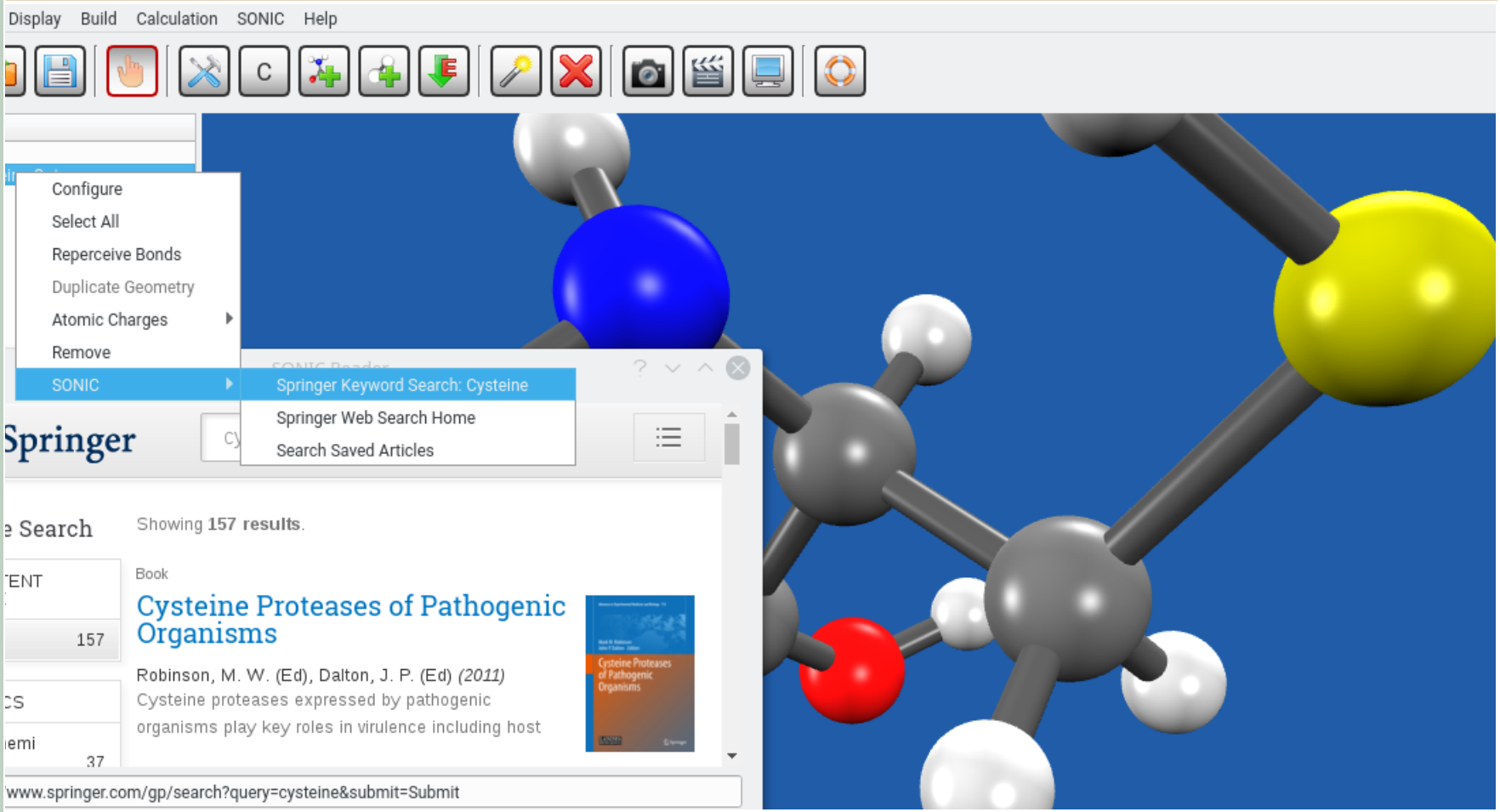
A3R applications may embed viewers for document formats such as e-Pub, HTML, and PDF; then supplement conventional publications with special components customized for individual manuscripts: e.g. (as in this case), a widget allowing readers to visually explore patterns in classical Indian music.



Linguistic Technology Systems

A3R Document Viewers as Embedded Components

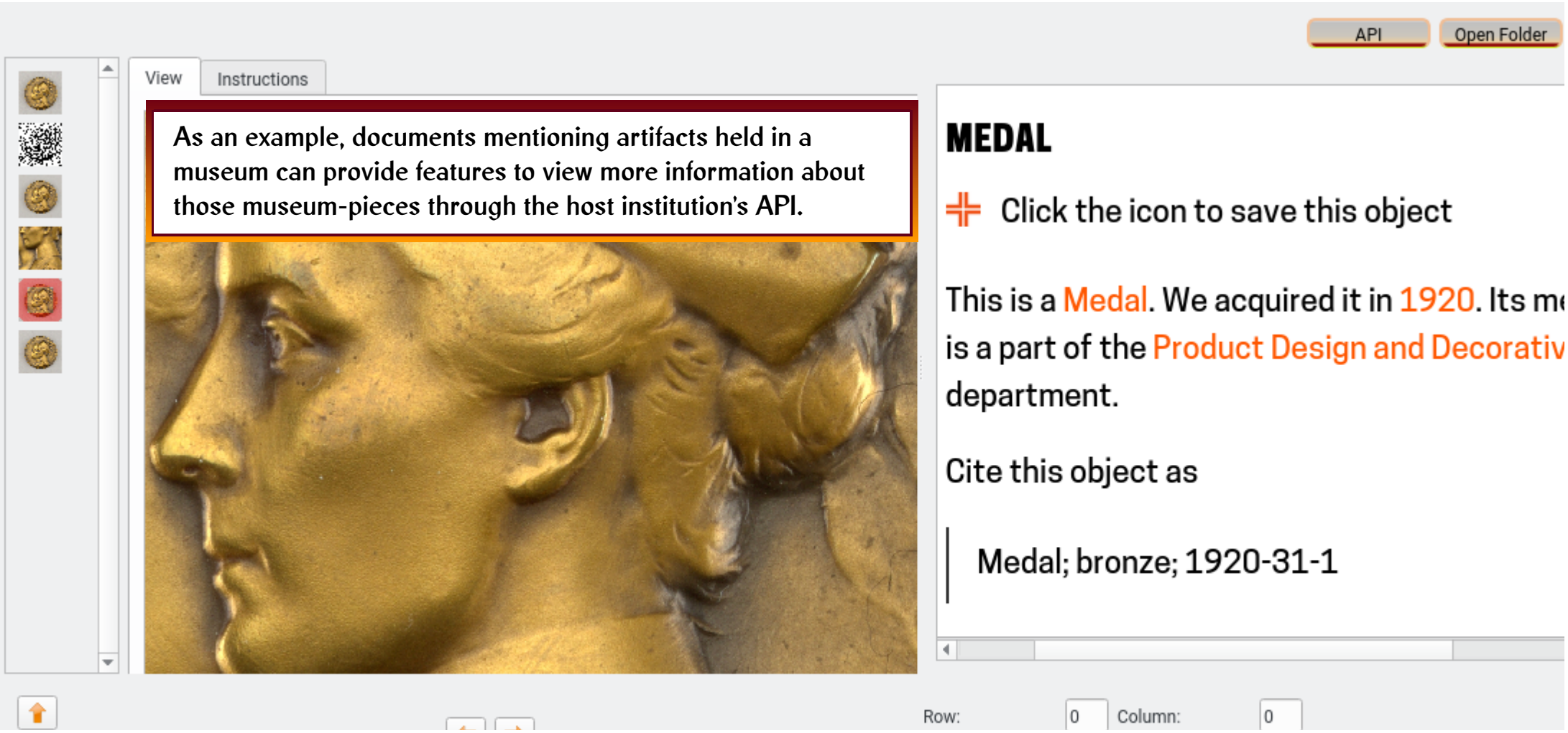
Document Viewers may also be embedded in host applications which provide domain-specific visualization capabilities. For example, chemistry papers might be viewed within IQmol (a Qt-based program for molecular visualization and physical/chemical analysis) via an A3R document-viewer plugin.



Linguistic Technology Systems

Document Viewers Augmented With APIs

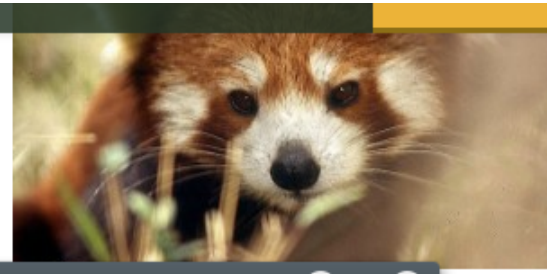
Another strategy for interactive publications is linking documents with APIs maintained by publishers, or by cultural or educational institutions.

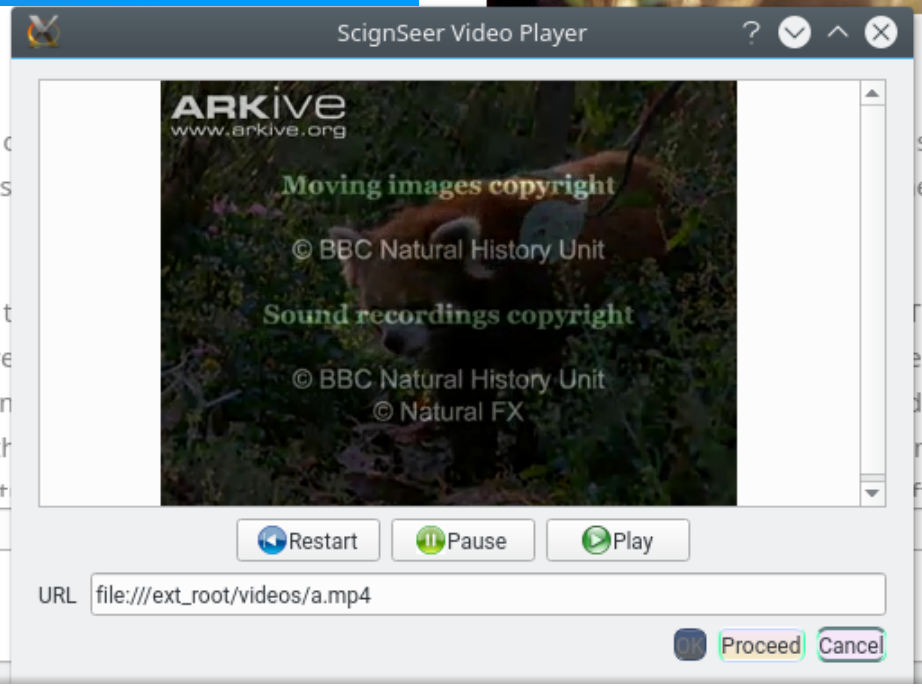


Embedded Multimedia

Custom-built A3R document viewers can provide convenient access to multimedia content embedded in or linked to texts — including audio files, videos, and 3D graphics scenes or models.

In this case a video player is launched in a dialog box, floating above the article text. For those reading digital books or articles, videos and other multimedia content can be presented through secondary windows launched via context menus; text and multimedia may thereby be viewed side-by-side.

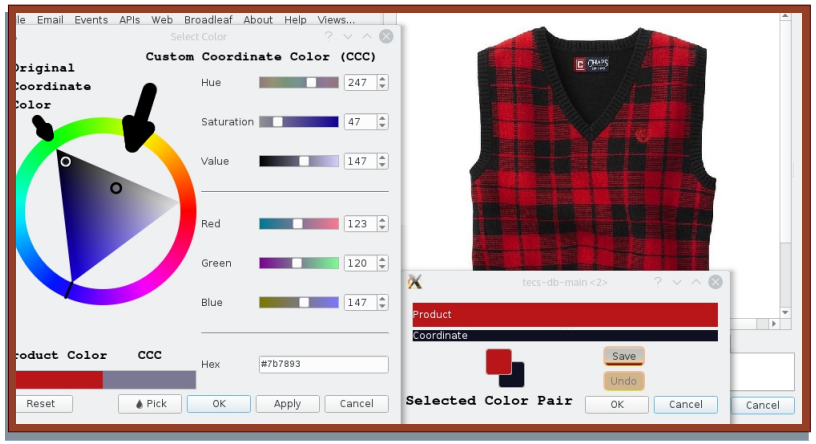


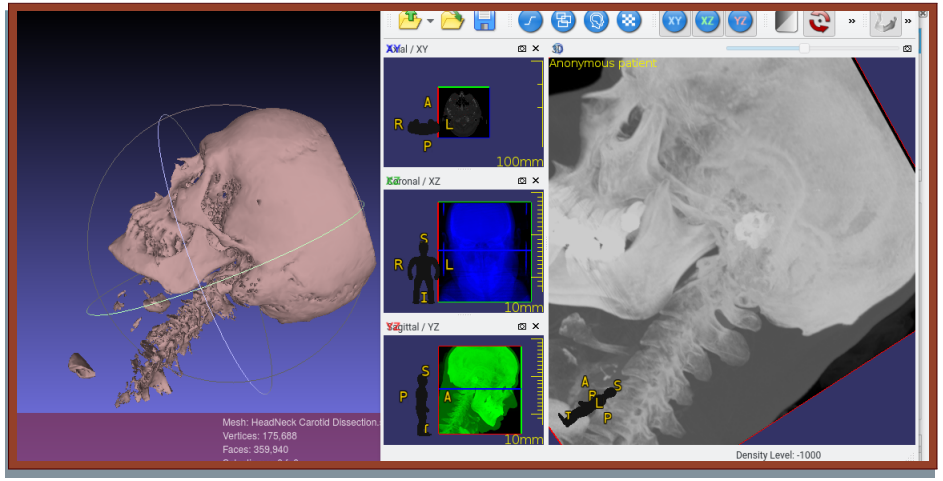


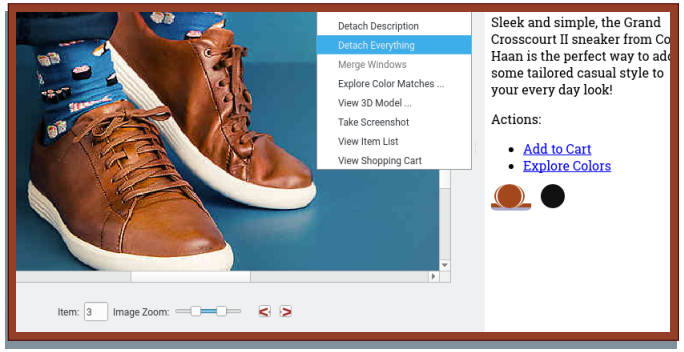
Linguistic Technology Systems

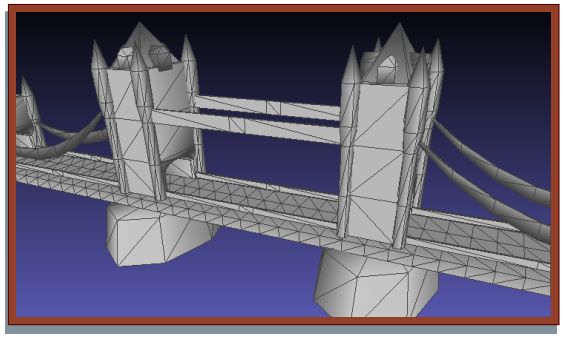
Thank You!

Please contact Linguistic Technology Systems for more information about dsC and/or other Software Development and Software Language Engineering Solutions: (917) 817-2184.









Linguistic Technology Systems