

# Getting Started with Tinderbox

**Mark Bernstein**

© Copyright 2015 by Eastgate Systems, Inc. All Rights Reserved.

Tinderbox™ is a trademark of Eastgate Systems, Inc. All other trademarks herein are used illustratively and are the property of their respective owners.

“Hill Academy” is a work of fiction. All persons, places, and institutions mentioned therein are imaginary, and any resemblance to actual places, institutions, or persons alive or dead, is entirely coincidental.

# Table of Contents

1. Getting Started With Tinderbox	2
2. First Notes	5
3. Links	26
4. Attributes	32
5. Prototypes	37
6. Actions	46
7. Sorting	49
8. The Attribute Browser	57
9. Aliases	63
10. Agents	65
11. Table Expressions	76
12. Timeline	82
13. Brainstorming With The Map	86
14. Brainstorming With Maps and Grids	101
15. Explode	109

# 1. Getting Started With Tinderbox

Tinderbox is the tool for notes, a unique and powerful approach to visualizing, organizing, and sharing information about complex problems.

This guide explores one way to get started taking notes with Tinderbox. If you'd like to know more about the design of Tinderbox and why it behaves as it does, you may enjoy *The Tinderbox Way*, a book about Tinderbox that is published by Eastgate. ([order here](#))

## **You don't need to read this**

There are lots of ways to use Tinderbox. The approach here is not the only or the best way to use Tinderbox. As you use Tinderbox, you will notice many alternate ways to do things; don't assume the method used here is the best for you.

## **Things Change**

This tutorial was originally written for Tinderbox 6.1. New versions of Tinderbox appear frequently, and the version you're using is probably newer.

As of this date, nearly 100 official Tinderbox versions have been released to customers.

Some details discussed here may vary from the version of the program you are using. Don't worry too much: the underlying principles and concepts still apply. If you're stuck, feel free to ask on the [Tinderbox Forum](#), or email [Eastgate](#).

## The Problem

In this tutorial, we'll be taking notes about the School Story – the literary genre about going to school that runs from *Tom Brown's Schooldays* through *Harry Potter*. We might be making these notes because we're planning to review a new novel, or because we're planning to write a school story of our own.

Other topics can be approached in exactly the same way. Perhaps you are studying nucleophilic substitution reaction rather than novels. Perhaps you're a private investigator, and you're assembling evidence about a shoplifting ring you have been investigating, or perhaps you're a political campaign director exploring how best to employ your volunteers.

Suspend your disbelief: If you're confronting problems that don't seem analogous to those described here, Tinderbox might still be able to help. Again, ask in the Tinderbox forum, where lots of Tinderbox users welcome such questions, or email Eastgate.

## Other Resources

Other places to find information about Tinderbox include:

Tinderbox Help, a detailed manual found in the Help menu.

The Tinderbox Forum, <http://eastgate.com/Tinderbox/forum/>, is an active and friendly meeting place for active Tinderbox users from a wide range of fields.

aTbRef – a Tinderbox Reference – is a technical reference for Tinderbox written by Tinderbox user Mark Anderson and available both as a web site and a Tinderbox document at <http://www.acrobatfaq.com/atbref5/> .

Mark Bernstein's book The Tinderbox Way discusses the core ideas that underlie Tinderbox. It is available as an eBook from Eastgate.

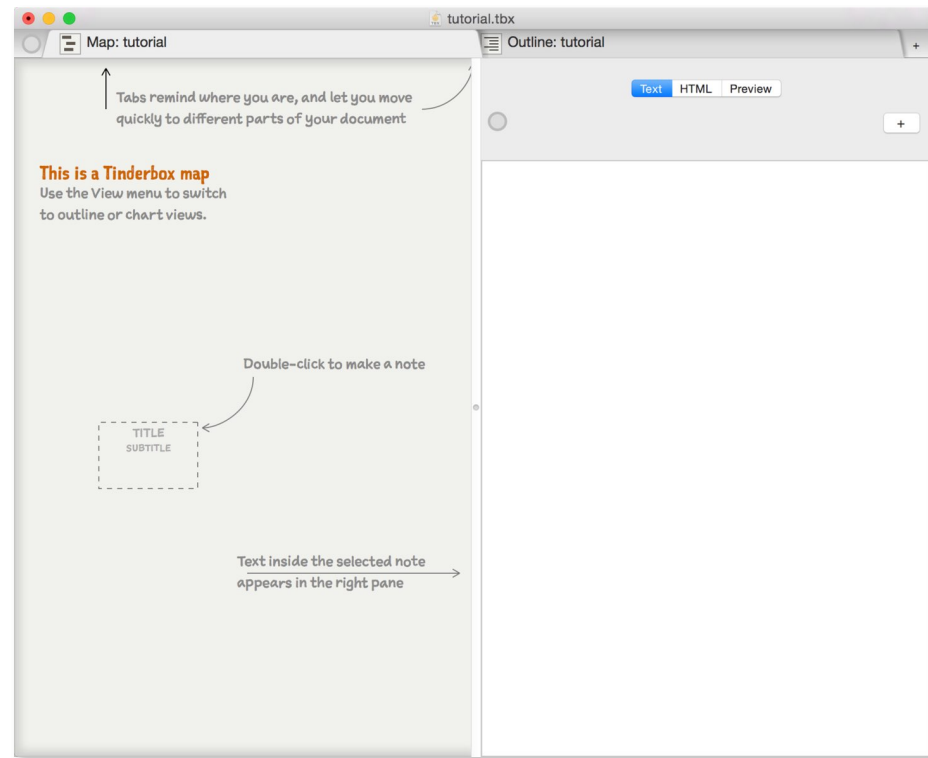
## 2. First Notes

### Install Tinderbox

You've probably installed Tinderbox already. If not, download your copy, open the disk image and drag Tinderbox into your Applications Folder.

Start Tinderbox by double-clicking the Tinderbox icon. Tinderbox will create a new, empty document.

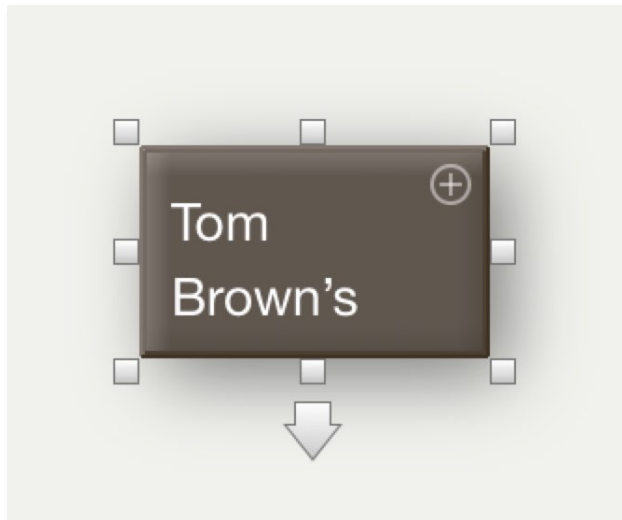
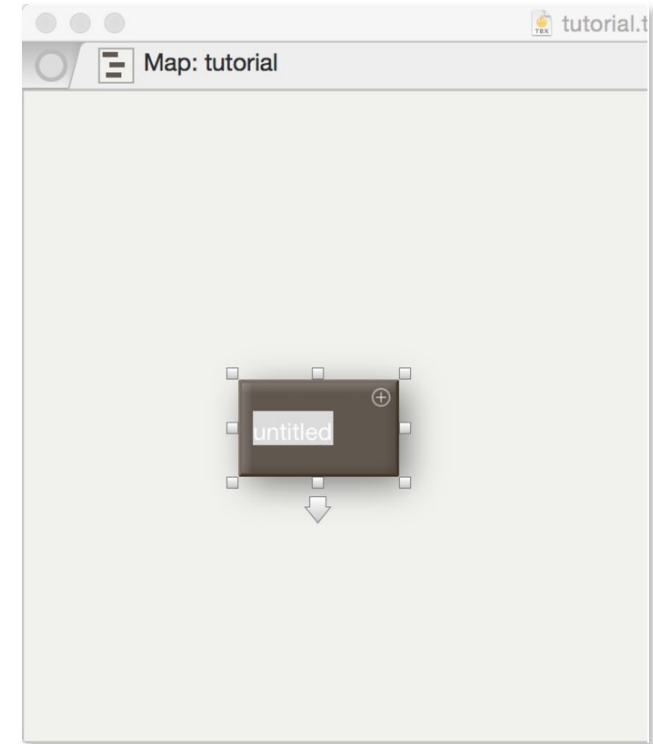
At the top of the Tinderbox window, you'll find the **tab bar**, which holds tabs for different views of your document. New documents contain two tabs – a Map view and an Outline view; you can add your own tabs as well. The **Map** tab is probably already selected; if it isn't, click the tab to select it.



## Make a note

Next, double-click in the center of the left window pane, which holds your new document's map view. Tinderbox will make a new note where you clicked.

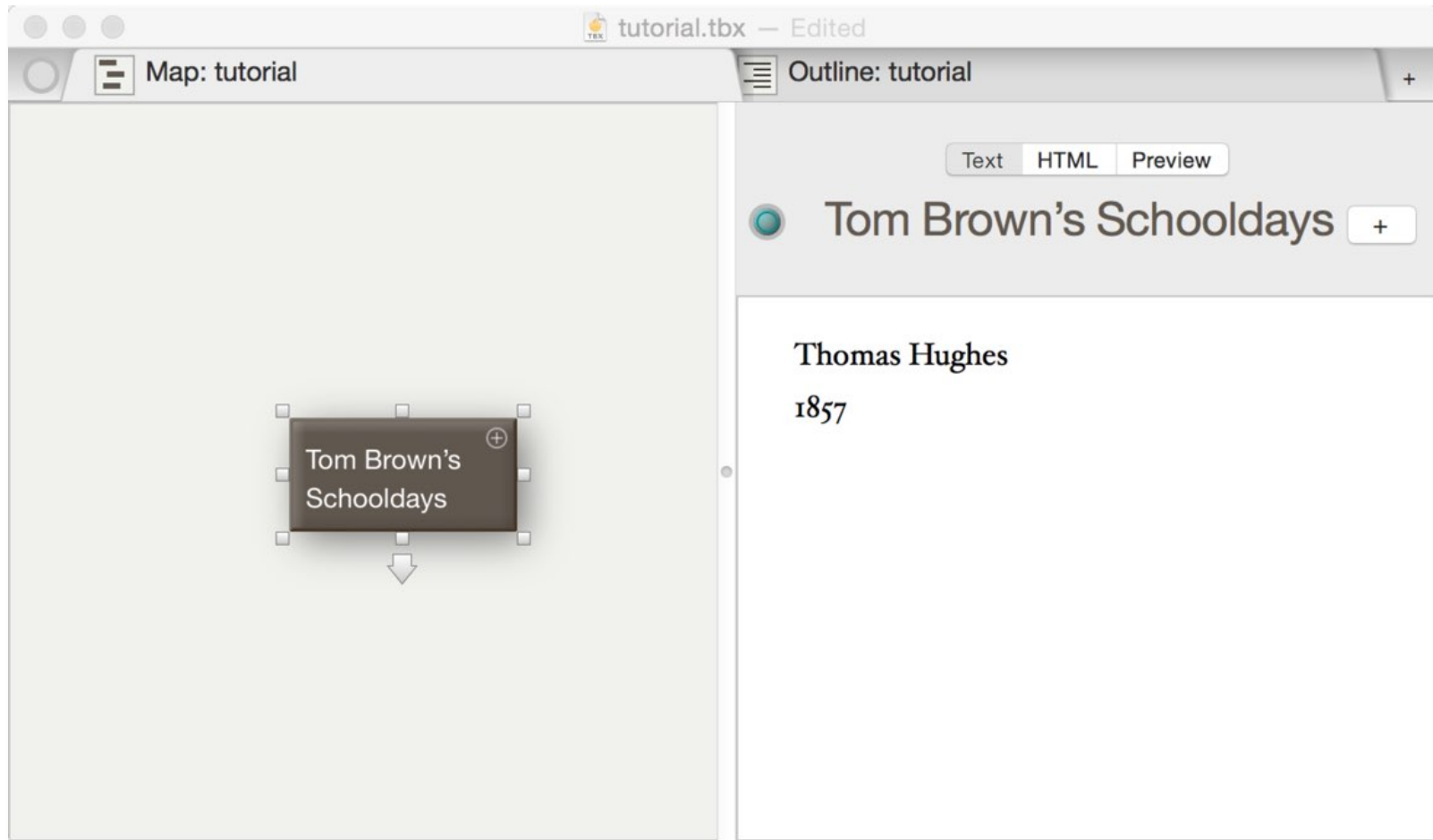
Each Tinderbox note has a title. Let's call this note *Tom Brown's Schooldays*. Type the title, and press **[return]** to finish creating the note.



The note is too small to display the entire title, so only the first part of the title is shown. That's often fine – some notes might have very long titles, and the first words may be all you require to identify them. But we can easily resize the note by dragging one of the resize handles until the full title is visible.

☞ If a note's selection handles are not displayed, click once on the note to select it.

- ☞ The Note menu provides options to **Expand Horizontally** or **Expand Vertically** until the entire title is visible, or to **Shrink To Fit** to choose a label font small enough to display the entire note.



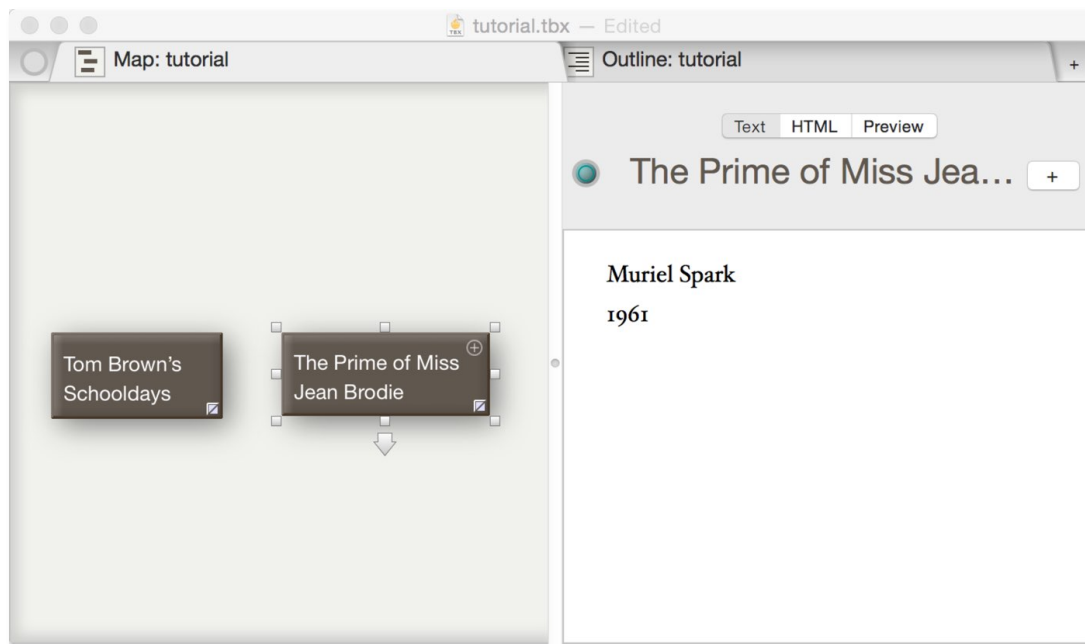
The right half of the Tinderbox window holds the **text pane**. In addition to its title, each note holds styled text. You can enter as much text as you like here, though short, focused notes are frequently the most useful. For now, we've noted the book's author and date of publication.



Move the divider that separates the map view and the text pane to adjust their relative sizes. The **Window** menu provides convenient shortcuts to let you view **Text Only**, **View Only**, or to divide the window evenly between **View** and **Text**.

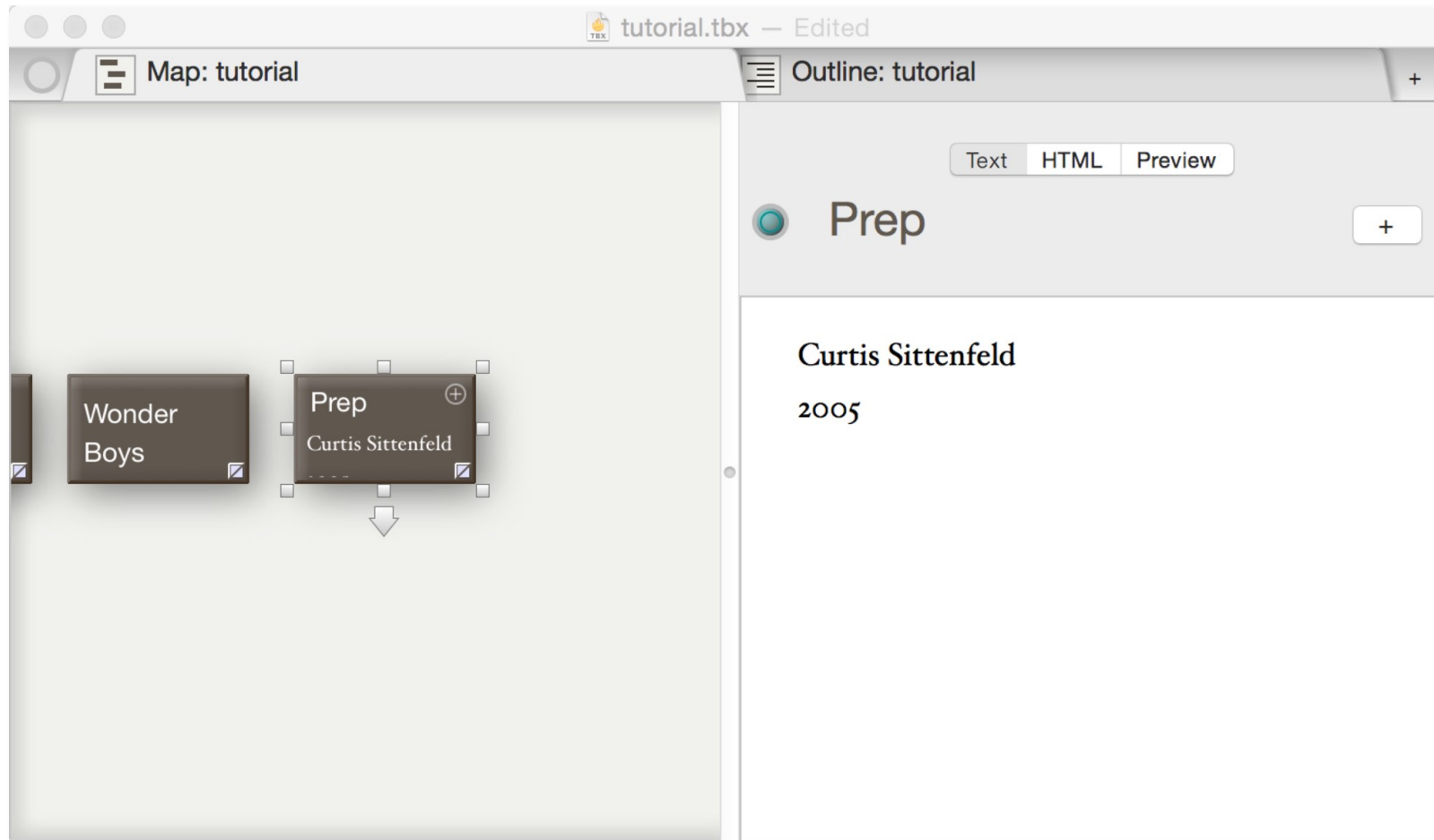
## Add More Notes

Returning to the Map pane, we can easily make another note. Simply double-click to the right of your first note, and create a note for *The Prime of Miss Jean Brodie*.



If you're in a hurry, you can also create a new note from the keyboard. Check to ensure that *The Prime of Miss Jean Brodie* is selected, and press **[return]**; Tinderbox will create a new note to the right of the selected note.

Name this note *Wonder Boys*. Press [return] to create another note, naming it *Prep*.



Note that, because the title is short, *Prep* displays part of the note's text beneath the title. Tinderbox displays the opening of each note's text if there's sufficient space; drag the bottom handles of each of the remaining notes until they are all approximately square.

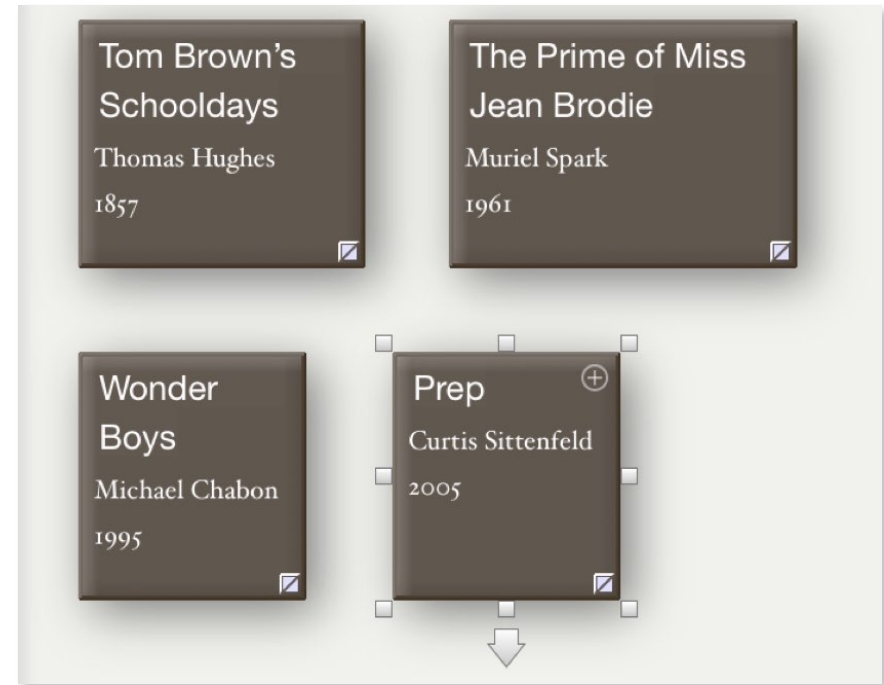
## Cleaning Up

We've built a long horizontal row of notes; this might be unwieldy.

Choose **Cleanup ▶ To Grid** from the View menu to arrange all the notes in the map in a compact grid.

You can move notes easily by clicking in their interior and dragging them in the map. As you drag, blue guidelines appear to help keep things neatly aligned. If you don't want these guidelines, you can turn them off in the View menu.

The View menu lets you **Magnify** or **Shrink** the map, or return it to **Standard Scale**. Using the trackpad, you can pinch-zoom to change the map scale. Two-finger double-tap the trackpad to return to standard scale.



## A Wall Of Notes

We've only begun to scratch the surface of Tinderbox, but already we have a powerful tool for capturing notes and organizing ideas. The Tinderbox map

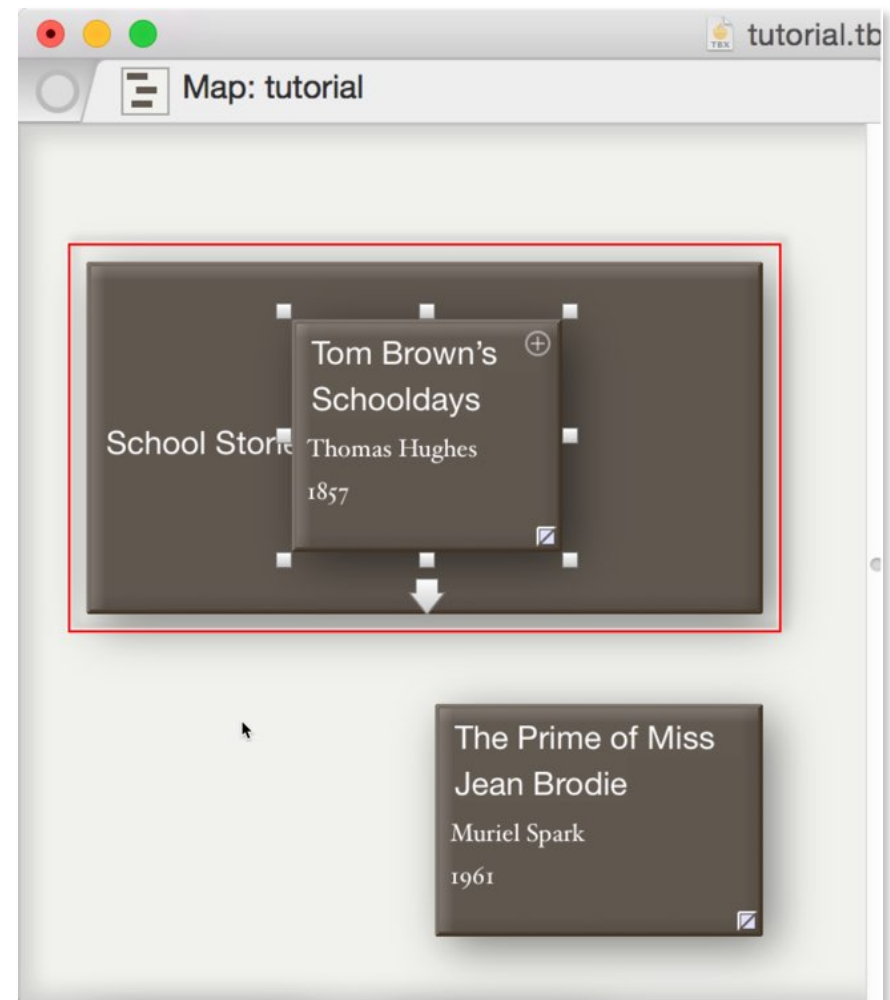
view presents, in essence, a huge wall on which you can arrange sticky notes however you please. The notes can be as concise or as long as you require, and each note can also hold as much text as you like.

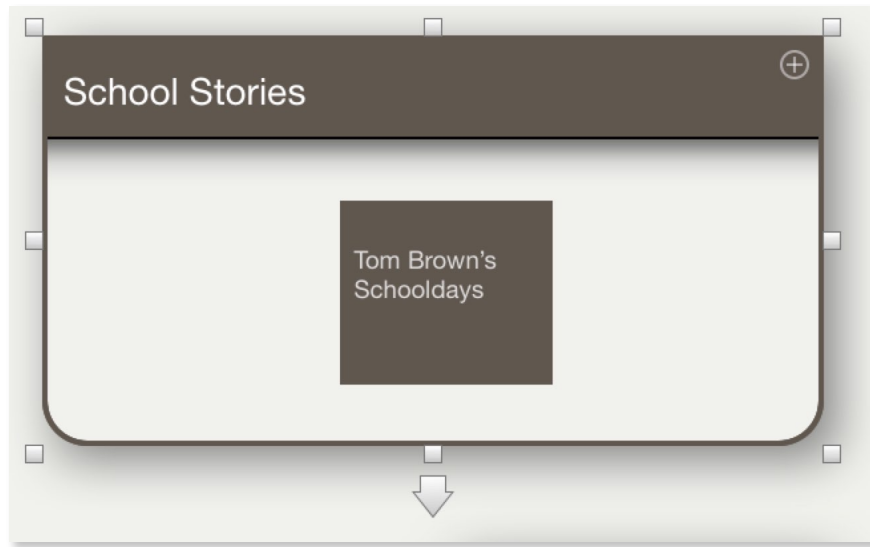
## Containers

A huge wall of notes can be a fine thing, but big projects may need more than one wall. Tinderbox *containers* are simply notes that hold other notes; each container becomes its own wall of notes. Containers themselves can hold other containers, which can be nested as deeply as you require.

So far, we've made four notes to represent four famous school stories. Next, we'll put them all into a container.

First, create a new note named School Stories. Expand that note so it's quite large.





Next, select *Tom Brown's Schooldays* and drag it over School Stories.

You'll see a red outline surround School Stories. Release the mouse button to place *Tom Brown's Schooldays* inside School Stories, which becomes a container. All containers have the same distinctive

shape, with a title bar at the top and rounded corners at the bottom.

Containers are also full-fledged notes: they have a title, text, and all the other properties that notes possess.

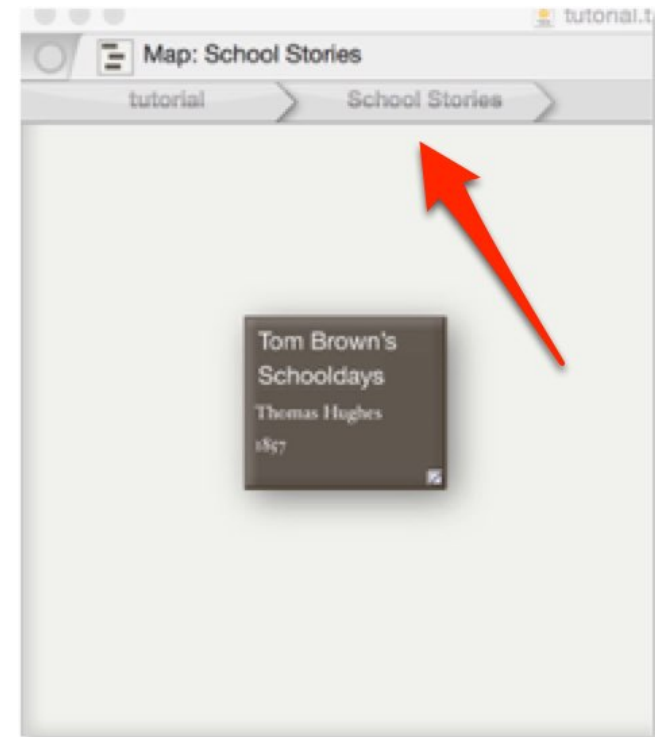
In map view, containers display part of their map, somewhat miniaturized, in the container's *viewport*. Much of the container's map may be outside the viewport, of course.

Now, drag each of the remaining notes into *School Stories*.

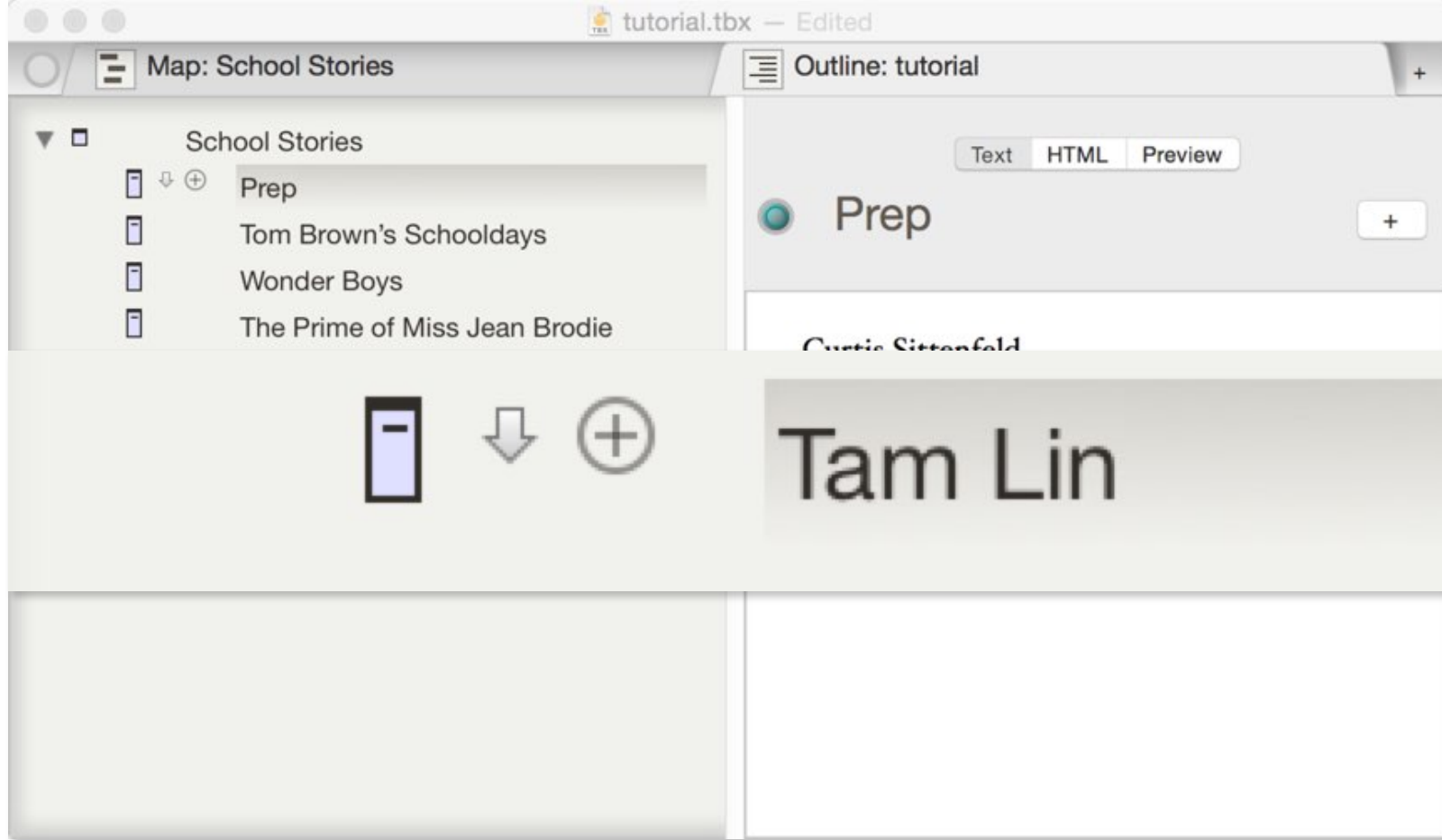
## Zoom Into A Container

To see the entire map inside a container, select the container and choose **Focus View** from the View menu. Or, simply select the container and press Down-Arrow.

When the current view is focused inside a container, the **breadcrumb bar** appears above the view window, immediately below the tab bar. Each nested container appears in the breadcrumb bar; here, we see that we're viewing the container School Stories which is inside the document named **tutorial**. Click on any breadcrumb to return the focus to an enclosing container.



👉 You can also press up-arrow to zoom up to the enclosing container.



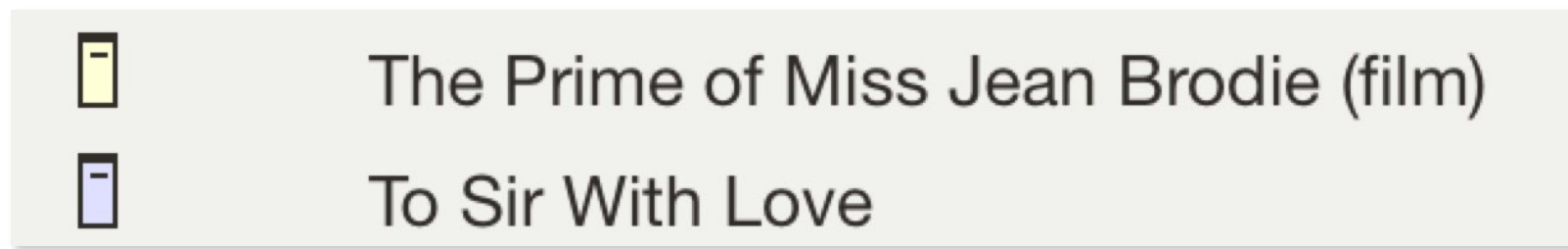
## Outline View

We can see a fresh perspective of our document by switching to the tab labelled **Outline: tutorial**. Here, we see the same document displayed as an expandable outline.

The container for School Stories appears at the top level, with its contents displayed beneath it. Clicking the **disclosure triangle** will collapse the corresponding section of the outline; clicking again will expand it.

Adding notes in Outline View is much like adding notes to a Map View. Select any note, and press **[return]**; Tinderbox will create a new note and insert it below the selected note. Name this new note *Tam Lin* for Pamela Dean's 1991 novel.

Notice the small rectangular icon that appears to left of each outline item. This icon packs a lot of information into a small space.



- The interior color of the icon reflects the time since the note was last modified. Newly-modified notes are light blue. Over the course of a day, the notes “dry out” to white, and then gradually yellow over the course of the subsequent year.
- Notes with no text have a short icon. Those with text have a tall icon, with one or more interior lines that represent roughly how much text the note contains.

You can drag notes in outline to rearrange them or to move them to new containers. In addition, selecting a note and pressing **[tab]** will indent the



note, if possible, placing it inside the preceding note. Pressing [shift-tab] will outdent the note, if possible, promoting it to become a sibling of its former parent.

## Talking About Hierarchy

Tinderbox containers form a hierarchical structure for your document. A note's container is called its *parent*. All the notes that share the same parent are *siblings*. All the notes immediately inside a container are that container's *children*; the first child in outline order is the *eldest child*, and the last child in outline order is the *youngest child*. We sometimes refer to a container's *ancestors* – its parent, the parent of its parent, etc. All the notes for which a container is an ancestor are *descendants* of that note. Every note in the document is a descendant of the document's *root note*. The immediate children of the document root are *top-level* notes.

## A Few More Children

Returning either to map or outline view, let's add a few more notes about school stories to the School Stories container. So far, all our notes involve

individual literary novels, but we may also want to consider some other things.

First, series of *dime novels* about school were commercially important in the early 20th century, and their memory shapes later novels. We'll add a single note to remind us about Ronald Reagan's favorites, the Frank Merrill Stories. More the 200 of these stories were published, but you don't want to enter 200 separate notes about 200 sentimental novels right away; we'll add one note for the present and we'll add more as needed – either as independent notes or as notes inside a Frank Merriwell container.



- ☞ When you are unsure how to represent a fact or concept, consider starting the simplest and easiest approach available. You can always return later and choose a more detailed representation.



families of badges and will display the badges of the family you select. Select the “?” badge from the Classic family, and click outside the popover to dismiss it.

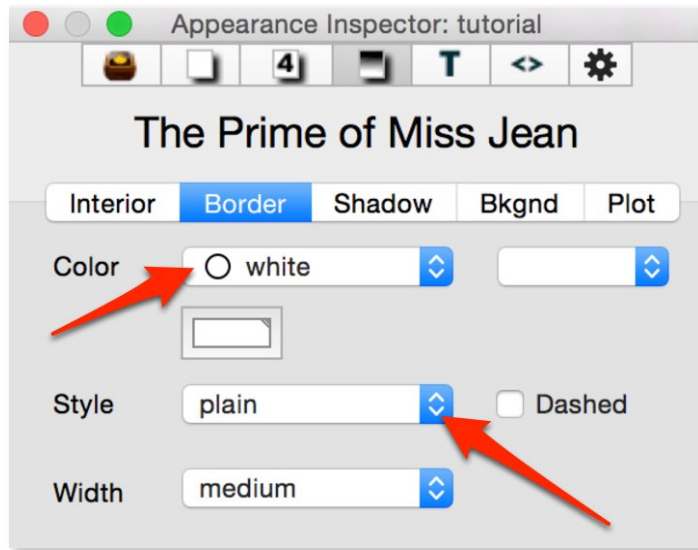
☞ Tinderbox comes with more than a hundred badges, but you can add your own badges, or even entire badge families.

## Other Ways To Mark Differences

Next, though *The Prime of Miss Jean Brodie* is an important and influential novel, it also inspired a memorable – but quite different – movie starring Maggy Smith with a screenplay by Jay Presson Allen. The differences between the novel and the film are

interesting and might deserve comment, and we will want in any case to be confident that, when talking about the book, we don’t accidentally cite the film. This issue will arise elsewhere, too – in E. R. Braithwaite’s *To Sir With Love*, for example (written and directed by James Clavell(!)), in R. F.





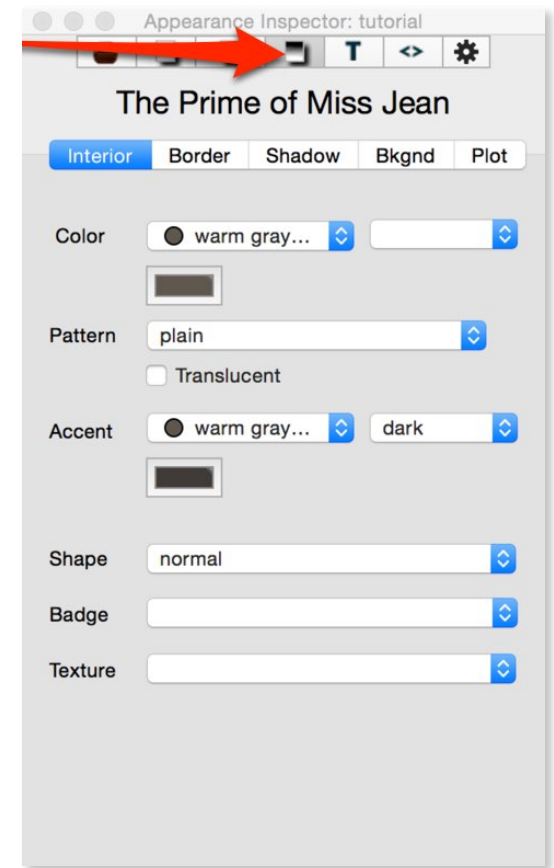
Delderfield's *To Serve Them All My Days*, and doubtless elsewhere.

Instead of using a different badge to distinguish films, we'll give notes about films a distinctive border.

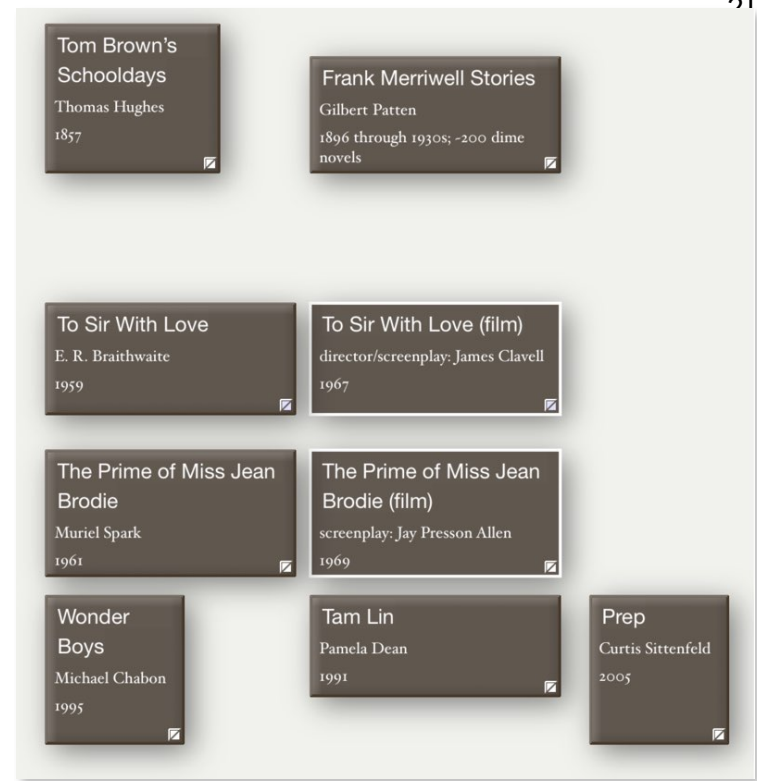
Select the note about this film, and open

the Inspector by choosing **Inspector** from the Window menu or by pressing ⌘-1. Select the **Appearance Inspector** — the fourth icon in the uppermost row of the inspector. Here, we can change the color of the note, the pattern used to draw its face, and the accent color used to draw that pattern. We can also change the note's shape, its badge, and we can apply a texture such as brushed metal or woodgrain.

In this case, we want to change the note's border, so select the Inspector's **Border** pane.



The initial value of the Border color will be “automatic” – Tinderbox selects the border color based on the color of the note. Instead, select a “white” border. The default style is also “automatic” which gives notes a bevelled edge. In its place, choose a “plain” border.



## Exploring With The Map

The Tinderbox Map is a flexible tool for exploring tentative or inchoate relationships in your notes. Even in the small collection of sources we've added here, we can begin to



find interesting hints of structure.

For example, two of our stories date from the late 19th or early 20th century. Drag these two notes together at the top of the map, and move any late 20th-century notes in that neighborhood lower down.

It's interesting (though perhaps unsurprising) that the older stories are sentimental while the later school stories are inclined to irony. The protagonists of the earlier works tend to be male, while the later stories often concern women chiefly or exclusively. Of course no conclusions can be drawn from such a small and arbitrary sample, but this simple bit of housekeeping does suggest relationships to watch and facets to explore, and perhaps might guide our further reading.

☞ You may want to resist the temptation to invariably line up your notes into neatly regimented grids. Informal associations and clusters often allow meaning to emerge gradually as your understanding of the problem grows.

# Adornments

Adornments are decorations in the background of maps. Adornments help to organize the space of the map and can remind you (and your collaborators) of how the map is currently organized.

Maps are not containers. A note is either inside a container or it is not; notes might lie partly inside an adornment and partly outside.

To add an adornment, right-click (or control-click) the map background where you want the adornment to appear and select **Create Adornment**. Drag and resize the adornments as you wish. Add two adornments to your map: one named “early stories (before 1940)” and the other named “1940-present”.





Adornments are translucent, and you can overlap adornments or place one adornment atop another. Adornments always lie behind all other notes, but one adornment might be in front of or behind another adornment. Because they're translucent, the ordering of adornments might not matter to you, but the Note menu does provide commands to send the selected adornment to send the selected adornment to the back, bring it to the front, and to move it up or down. Using the inspector, you can change the color, border, and other appearance properties of an adornment just as you can change a note. If you wish, you can even add a textual note to an adornment.

## Summary

At this point, we can create notes and containers in maps and outlines. We can move the notes to build conceptual clusters in maps and to construct hierarchical lists in outlines. We've added badges to remind us to reexamine specific notes, and chosen a distinctive appearance to distinguish a group of related notes.

This gives us a simple array of note-making spaces, like large walls of sticky notes or large tables covered with note cards. Everything can be searched,

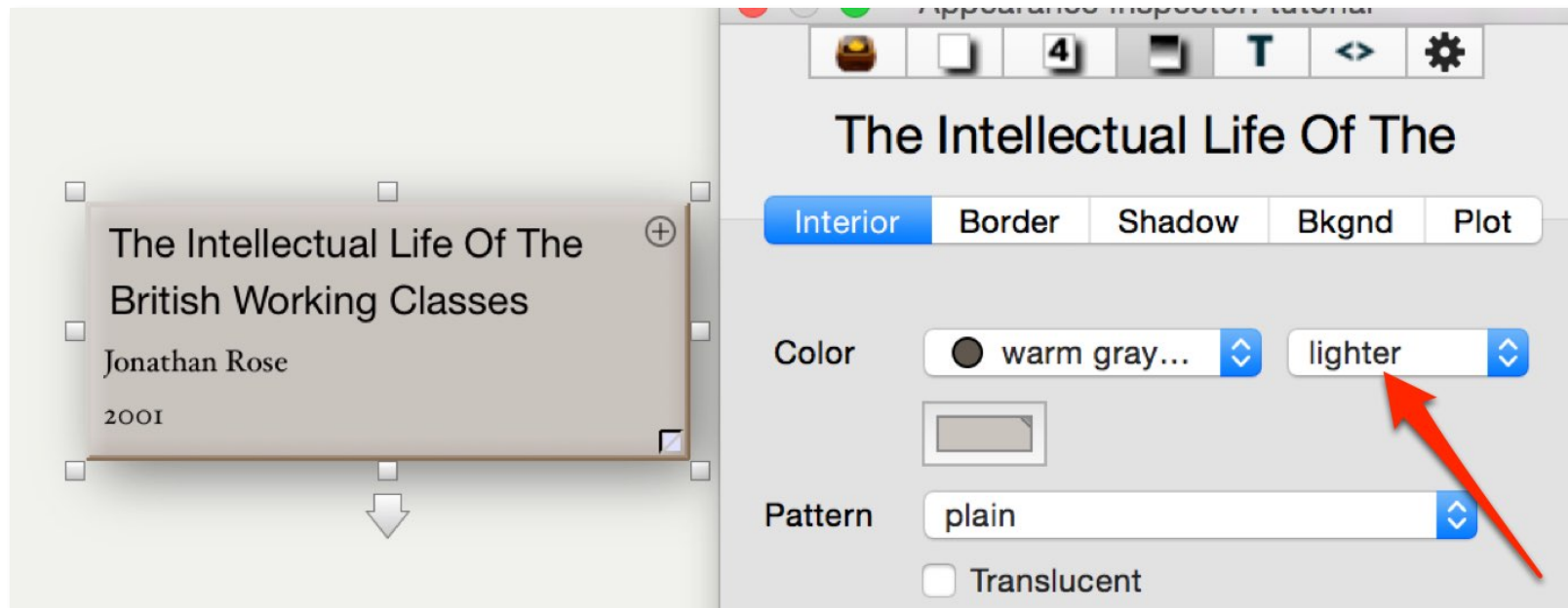
too. Though we're not yet using much of Tinderbox's capabilities, we already possess a powerful environment for exploring complex ideas.

At this point, we rely on discipline for representational accuracy: if we add a note about a film, we have to remember that films have white borders. That's not always a bad approach. But Tinderbox provides tools that can help automatically suggest or enforce constraints, tools that help your document keep itself better organized. In chapter 5, we'll explore Prototypes, the most important of these tools.

### 3. Links

#### A Horse Of A Different Color

Our research may not be confined to the school stories themselves; we might, for instance, consult what other people have written about the school story. For example, Jonathan Rose has a wonderful chapter in *The Intellectual Life Of The British Working Classes* on working class enthusiasm for the far-from-working-class *Tom Brown's Schooldays* and its successors. So, we'll want to add a note for this source, too.



Because it's not itself a school story, we want to distinguish this note. Select the note, open the Appearance inspector if it's not already selected, and change the note's color from "warm gray dark" to **lighter** warm gray dark by choosing "lighter" from the color modifier popup menu.

Tinderbox frequently uses symbolic color names like "red" or "lighter warm gray" because this makes it easy to assign identical colors to different notes at different times. You can change just what hue these symbolic colors represent at any time.

## Adding a Link

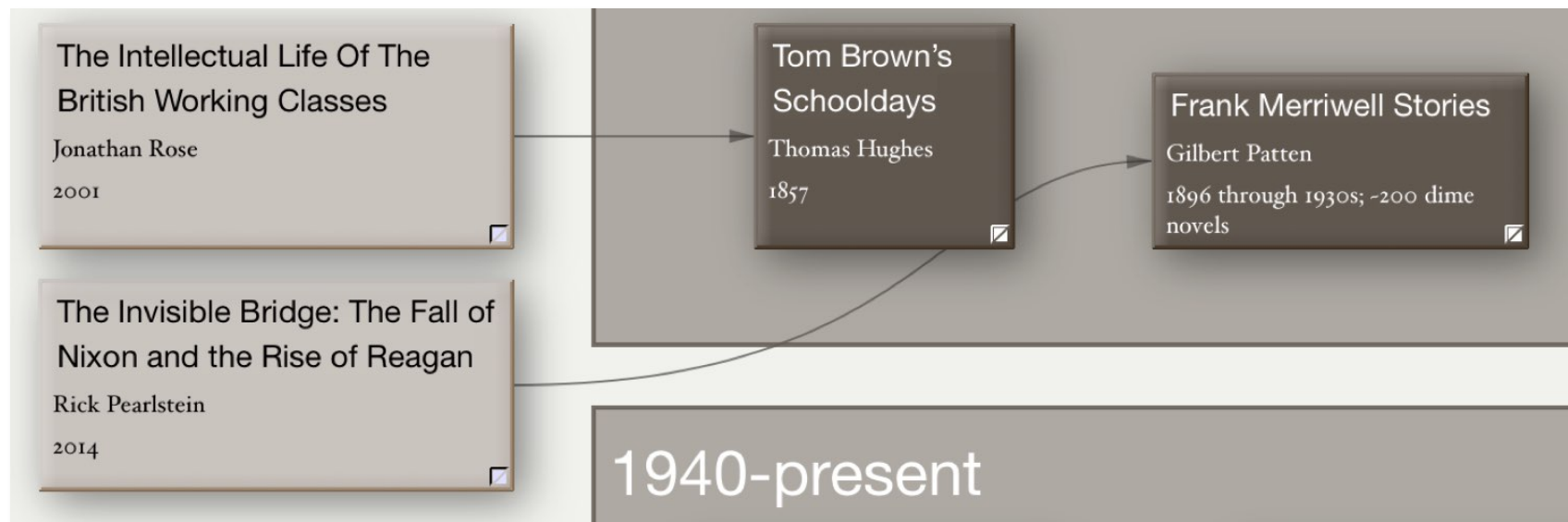
We added this book because it's a useful source for understanding stories like *Tom Brown's Schooldays*, so it makes sense to place it near *Tom Brown*. But what are we going to do if a source mentions several stories – and those stories appear in different places in the map?

We can also represent the connection between the discussion of the story and the story itself with a *link*. To make a link, simply select a note and drag a link



out of the *link widget* arrow beneath the selected note. Drag the link to the destination, and release the mouse button; a popover will offer to create a link.

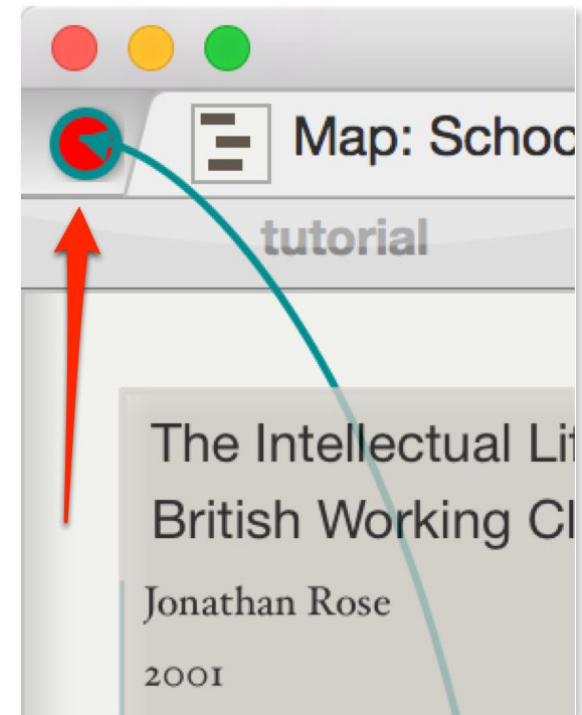
Links will remain attached to their notes as you move them in the map or reorganize them in the outline. In map view, links are drawn between items that appear in the same map; when a note links to another note elsewhere in the document, a small outbound arrow, called a *link stub*, is drawn beneath the source. When a note outside the map links to a note, the link is drawn as a small inbound link stub above the destination.



Rick Pearlstein discusses the Frank Merrill stories and their lasting impact on Ronald Reagan in *The Invisible Bridge: The Fall of Nixon and the Rise of Reagan*. So we'll create a note for that book and link it to Merriwell.

## The Link Parking Space

So far, we've been able to make links simply by dragging the link from the link widget to the link's destination. At times, though, the destination might not be conveniently nearby. In that case, we drag the link into the **link parking space**. Release the mouse, and then locate the destination note – perhaps scrolling the map, or even using a different tab. Then, drag the link from the parking space to its destination, release the mouse button, and create the link in the usual way.

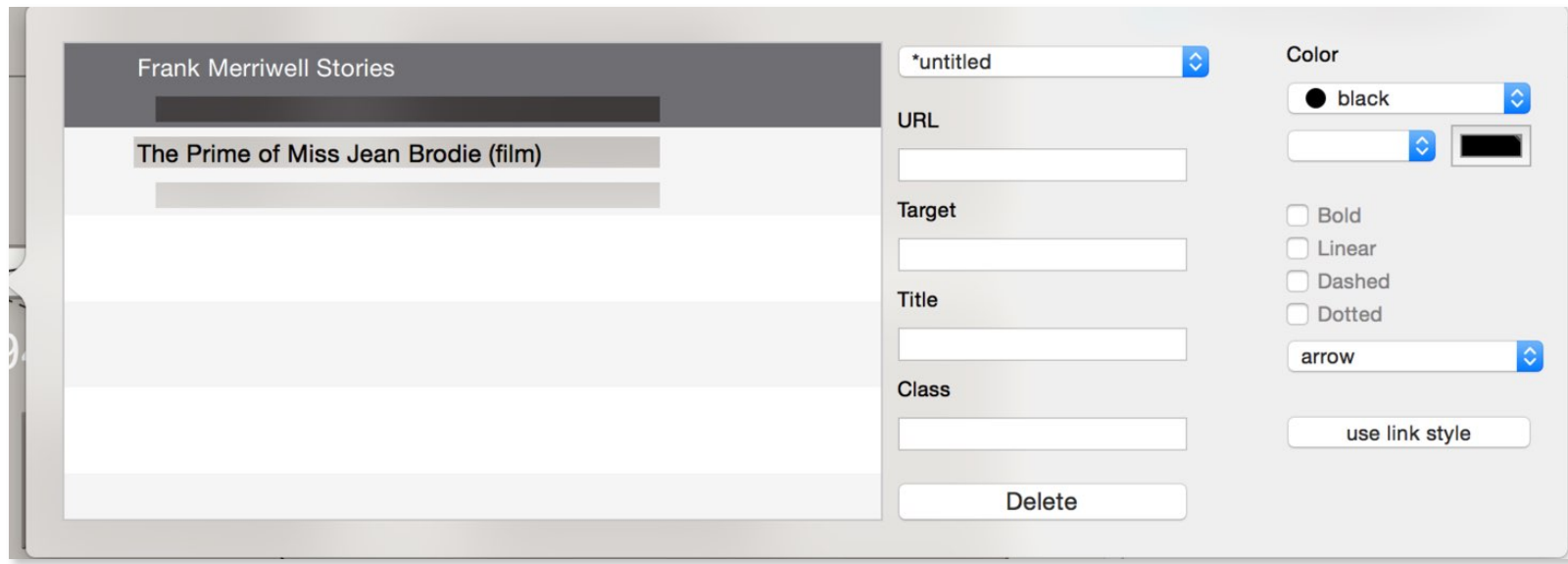


## Browsing Links

If a note has outbound links, selecting the note will highlight those links. For additional information



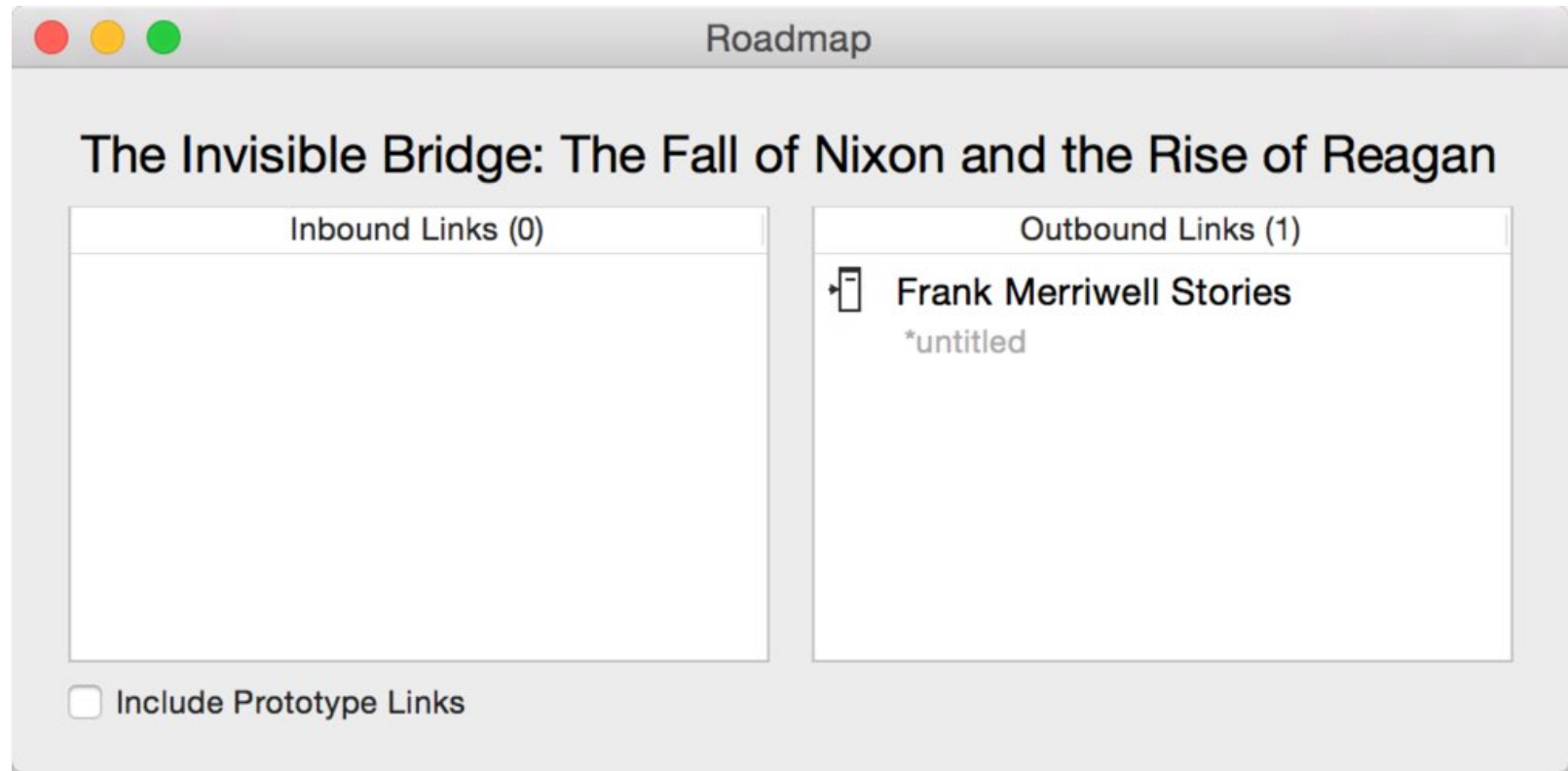
about the links, choose **Browse Links** from the View menu, or click on any of the highlighted links' Information buttons. The Browse Links popover permits you to examine each link, and also lets you modify the link's appearance – making to bold, or drawing it in color.



## Roadmap

The **Roadmap** view lets you examine which links arrive at and depart from the selected note. Double-click a note in either list to select it, refocusing the roadmap on the link's source or destination.

Note that the Roadmap is a tear-off popover. If you drag the background of the popover, the roadmap will be converted to a standard window and will remain open until you close it or Quit Tinderbox.





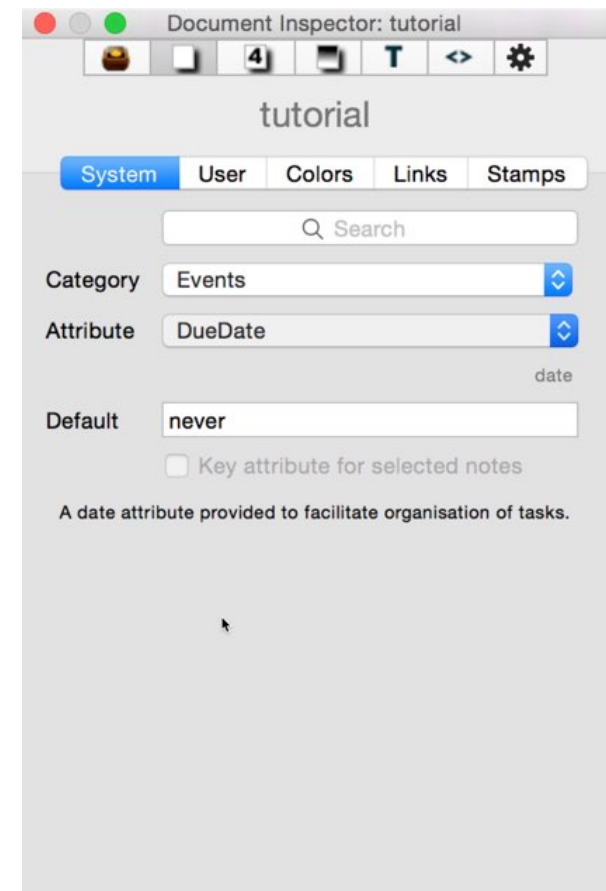
## 4. Attributes

In addition to its title and text, each Tinderbox note holds a long list of attributes. Several hundred *system attributes* are built into each Tinderbox document and have specific meanings to the system; for example, the attribute **Color** is the dominant color of a note, and the attribute **WordCount** is the word count for that note's text.

☞ To prevent confusion, we sometimes preface an attribute name with a “\$” to be clear that we’re discussing an attribute. For example, **\$Color** controls the note’s color.

To make them easier to use, system attributes are organized into *categories* of related attributes. For example, **Map** attributes include **\$Height** and **\$Width** – the dimensions of the note when displayed in map view. **Event** attributes include things like **\$StartDate**, **\$EndDate**, and **\$DueDate**.

The System pane of the Document Inspector



allows you to examine each system attribute in the document. First, select a category, and then select the attribute of interest – or use the Search area. The Inspector will then display:

- The **type** of the attribute's value – whether the attribute holds a string, a number, a color, or a date.
- The **default value** of the attribute – the value used if not other value has been specified. For example, the default value of \$Color in a new Tinderbox document is “warm grey dark”.
- A brief description of the meaning of the attribute.

## Key Attributes

You can use attributes to store specific information about your notes. For example, we've been noting each story's author and publication date in the text. We could instead note this information in attributes.

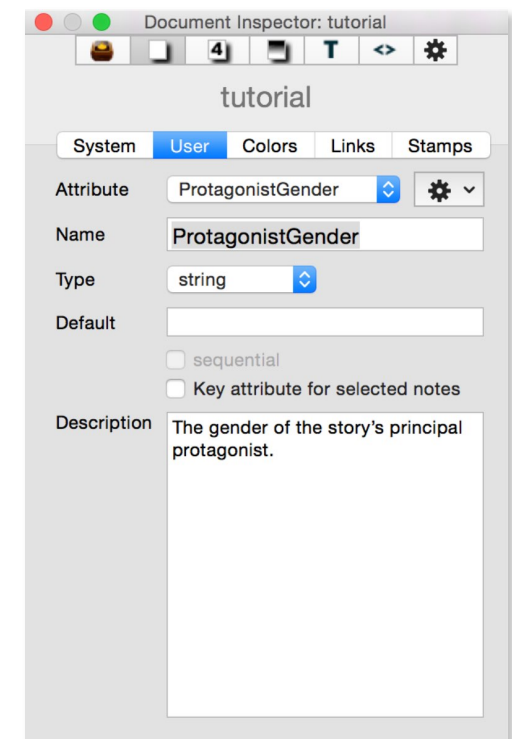
Each note's **key attributes** are those attributes which we regard as particularly useful for that note – so useful, in fact, that we'd like to display them whenever we inspect that note.

To add some key attributes to *Tom Brown's Schooldays*, select the note and then press the *key attributes button* – labelled with a “+” sign – in the upper right-hand corner of the text pane. A popover called the *key attribute picker* appears to let you select which attributes are of particular interest. Scroll down to the **Reference** category, select it, and choose **Authors** and **Publication Year** from the list of attributes. Finally, click outside the popover to dismiss it.

Now, double-click in the right half of the Authors line, enter “Thomas Hughes” (or copy it from the text), and press [return]. Double-click the right half of the Publication Date line, enter “1857,” and press [return]. Now, whenever you select *Tom Brown's Schooldays*, this small table of useful information will appear above the text.

## User Attributes

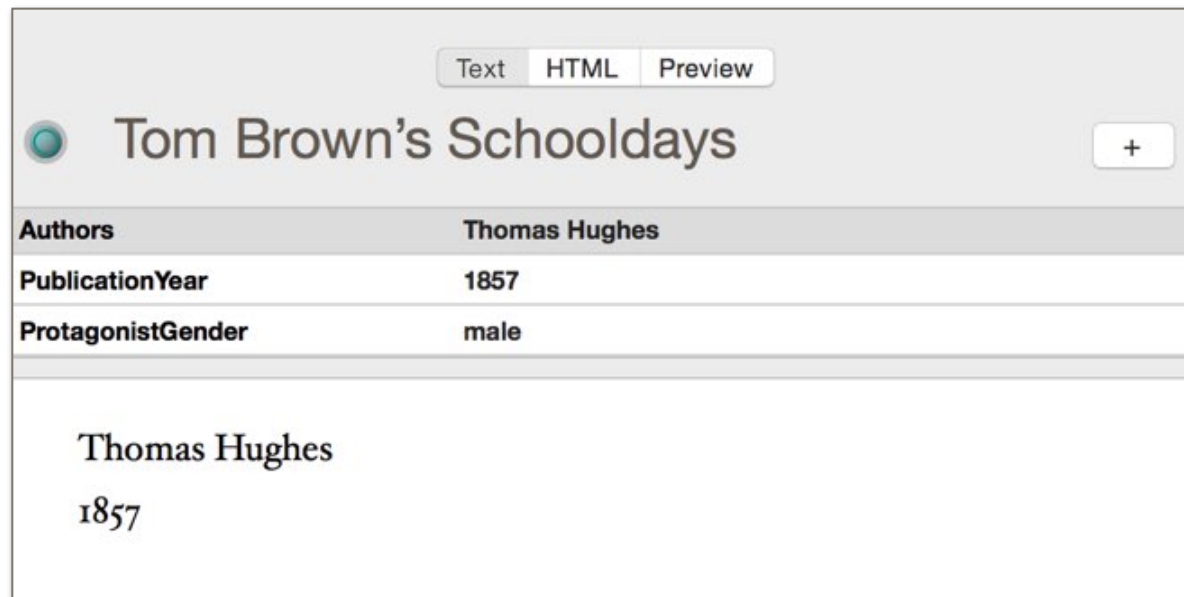
It happens that Tinderbox offers built-in attributes for Authors and PublicationDate, so we can use those right away. We can also create additional attributes tailored to the work at hand. We call these *user*



*attributes*, and create them in the User pane of the Document Inspector.

To create a new user attribute, click on the action button (labelled with a gear) and choose **New Attribute** from the pulldown menu. Choose your attributes name, its type, and its default value. You may also enter a brief description here, to remind yourself and your collaborators what the attribute represents and how you want to handle special cases.

In your document, define a string attribute `$ProtagonistGender`, and add it as a key attribute for *Tom Brown's Schooldays*.



The screenshot shows the Document Inspector interface for a document titled "Tom Brown's Schooldays". At the top, there are tabs for "Text", "HTML", and "Preview". Below the title, there is a table with the following data:

Authors	Thomas Hughes
PublicationYear	1857
ProtagonistGender	male

Below the table, there is a preview of the rendered HTML, showing the text "Thomas Hughes" and "1857" on separate lines.

## Don't Go Overboard!

New Tinderbox aficionados are sometimes tempted to create dozens of user attributes right away. This is sometimes the right way to start, but often it's a mistake – especially at the outset of a challenging, long-term project. It can be difficult to predict exactly what information you'll absolutely need to capture, and so you might find yourself spending time recording details that you won't actually need.

Because you can add new attributes later, it's often best to begin by formally formalizing only the information you are confident you'll want later. Record everything else in the text (or in subsidiary notes); later, if you decide that an attribute is warranted, you can easily add it just as we added `$ProtagonistGender`.

# 5. Prototypes

## Adding Notes: Duplicate and Modify

As our reading continues, we will naturally add new stories to our notes. Now that we're using more elaborate notes, though, each new note requires a little extra work:

- add key attributes for Authors, PublicationDate, and ProtagonistGender
- change Color if the note is a secondary source
- change BorderColor and BorderStyle if the note is a film

One way we can save a little time when adding a note is to locate a similar note, duplicate it, and then enter the new information.

For example, we'll add a note about R. F. Delderfield's *To Serve Them All My Days*.

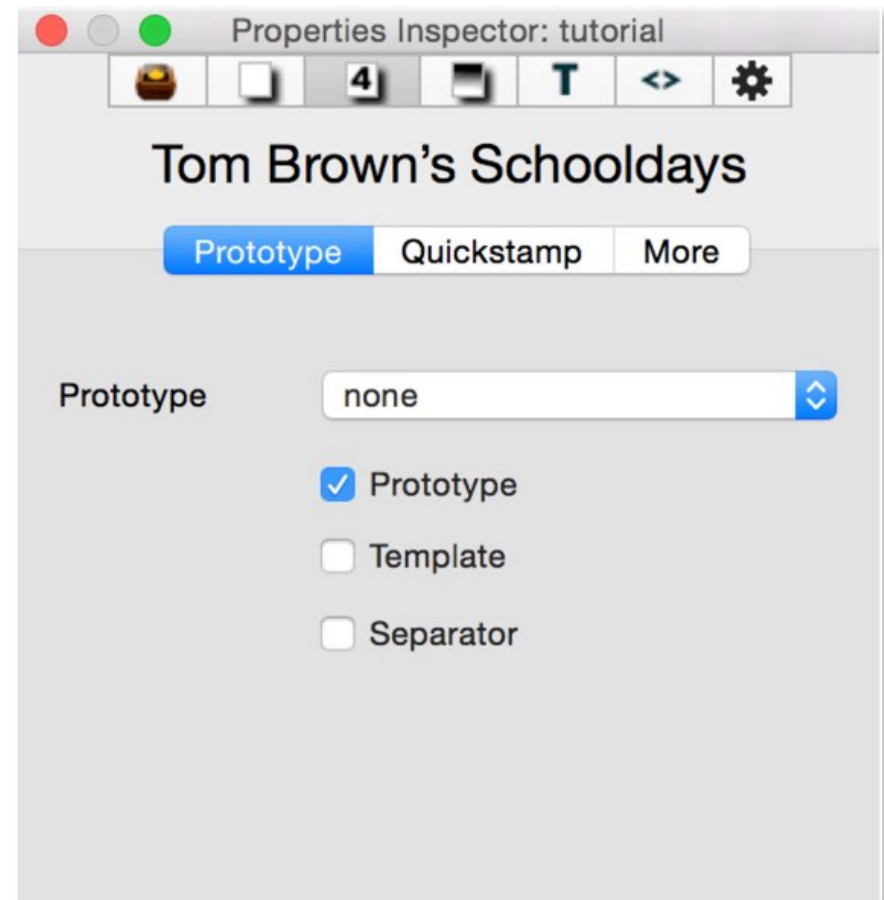
1. Locate and select *Tom Brown's Schooldays*.

2. Choose **Duplicate** from the Edit menu. (Don't confuse this with File ▶ Duplicate, which makes a duplicate of the file.) Tinderbox creates a new note titled *Tom Brown's Schooldays copy*.
3. Rename the copied note, *To Serve Them All My Days*.
4. Change the Author to "R. F. Delderfield", the PublicationDate to "1972," and the ProtagonistGender to "male."
5. Drag the new note to a suitable place in the map view.

Similarly, when creating a note about the 1980 BBC miniseries of *To Serve Them All My Days*, we can save a few steps by duplicating the film of *The Prime of Miss Jean Brodie* and then changing the pertinent fields.

## Adding Notes: Prototypes

Rather than copying notes, we can instruct Tinderbox that our new note



should use some other note as a *prototype* – that the new note should emulate some other note in all ways except where we tell Tinderbox otherwise.

Any Tinderbox note can serve as a prototype for other notes. Let's use *Tom Brown's Schooldays* as a prototype for school stories.

1. Locate and select *Tom Brown's Schooldays*.
2. Open the Inspector and select the Properties Inspector.
3. Choose the Prototype pane
4. Check the “Prototype” checkbox.

Next, let's create a note for Dorothy Sayers' *Gaudy Night*.

1. Double-click in the map to create a new note. Name it “Gaudy Night”.
2. Beneath the selected note, you'll see a Prototype tab.





3. Click and hold the Prototype tab, and select **Tom Brown's Schooldays** from the menu of prototypes.

The new note now inherits attributes from its prototype unless we've assigned specific values for that note. Thus, Gaudy Night now inherits the key attributes currently used by Tom Brown's Schooldays. Change the Publication Date of Gaudy Night to 1936, the Authors to "Sayers, Dorothy," and note that its principal protagonist is female.

Similarly, select *The Prime of Miss Jean Brodie (film)* and use the Inspector to allow this note to serve as a prototype. Make a new note for Rian Johnson's film *Brick* (1985), and use *The Prime of Miss Jean Brodie (film)* for its prototype. Notice how the new note inherits the border from its prototype, which we want. The new note also inherits the prototypes text, which we don't want: replace it with the appropriate Screenwriter (Rian Johnson) and date (1985).

## How Inheritance Works

Notes retain a connection to their prototype, and use the prototype's current value whenever the note itself has no specific value for that attribute. For example, select Tom Brown's Schooldays and open the Appearance

Inspector. Change its color to “red”; not only does Tom Brown’s Schooldays turn red, but Gaudy Night also turns red. Change the color back to “warm gray dark”; both notes return to their former color.

If a note has a value for the attribute, Tinderbox uses that value.

If a note has a prototype, and the prototype has a value for the attribute, Tinderbox uses that value.

If the prototype itself has a prototype, and *that* prototype has a value for the attribute, Tinderbox uses that value.

If the note has neither a value for the attribute nor a prototype, Tinderbox uses the default value for that prototype.

If a note has a specific value for an attribute, that value overrides inheritance. Assigning a prototype for a note means “make this note just like its prototype, except when I tell you otherwise.”

**The Fine Print:** A few attributes are not inherited, because inheritance would make no sense for them. For example, every note has its own map position \$Xpos and \$Ypos, so these attributes are not inherited. Some read-only attributes like \$WordCount and \$Modified aren’t inherited either.

Prototypes can use prototypes themselves, and you can use as many prototypes as you like. Resist the temptation to go overboard in creating

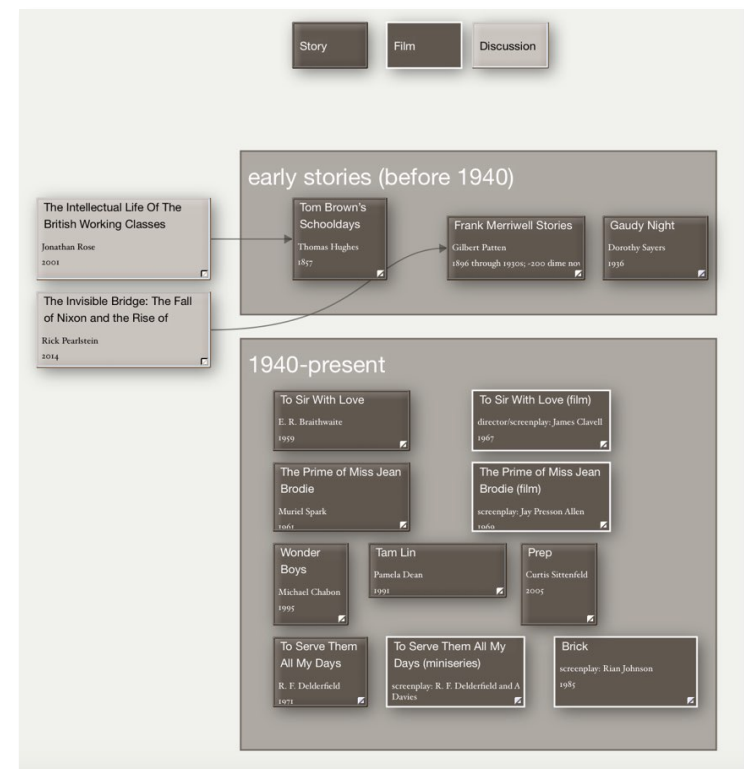
dozens of prototypes at the outset of a project; many Tinderbox projects only require a handful of prototypes.

## Dedicated Prototypes

Though any note can serve as a prototype, many people prefer to create specific notes simply to serve as prototypes for other notes. For our current work, we might create three prototypes: Story, Film, and Discussion.

First, make a Duplicate of Tom Brown's Schooldays and drag it to the top of the map. Name it "Story", and remove the values for Authors, PublicationDate, and Protagonist Gender. Using the Properties Inspector, allow "Story" to be a prototype, and also uncheck "prototype" for Tom Brown's Schooldays

Next, make a Duplicate of "The Prime of Miss Jean Brodie (film)". Drag the duplicate next to the "Story" prototype, and rename the new duplicate "Film." Again, use the Properties Inspector to



allow Film to be a prototype and to remove the Prototype checkbox from The Prime of Miss Jean Brodie (film).

Finally, create a new note named Discussion near the prototype Film. Let Discussion use Story as a prototype, and use the Inspector to allow Discussion itself to serve as a prototype. Change its color modifier to **lighter**, so its color matches the color of the other secondary sources.

Now, select each story in turn and set its prototype to Story. Select each film, and change its prototype to Film. Select each secondary source, and change its prototype to Discussion.

- ☞ If you like, you can shift-select several notes and change their prototype at the same time.
- ☞ The Prototype Tab of a selected note displays the name of the note's prototype, if it has one.

## Summary

The first key to making useful notes is, simply, to make notes – to write stuff down. Over the long haul, this requires striking the right balance between capturing information quickly and capturing the information you need. Prototypes help speed note-taking by (a) setting appropriate initial values

automatically, and (b) highlighting key values to remind you to record essential data.

For example, a restaurant reviewer's notes might include almost anything: a restaurant's decoration, the history of food writing, the best way to prepare fresh beets, the condition of women in the professional kitchen. Still, a newspaper reviewer must capture specific data that are required for a review: serving hours, phone number, credit card policy. It can be easy to overlook these mundane requirements in the face of more interesting questions; prototypes can help remind you to capture boring information you will need.

Nevertheless, avoid the temptation at the outset of a new project to create dozens of prototypes, each with its own slate of attributes. No one loves to fill out long forms – especially not when time is short and important ideas need to be captured immediately. Each new prototype will present you with new choices and new doubts whenever you create notes: does the note you're about to create actually fit the prototype? We have already encountered this in our tiny project: the prototype describes “story” but one note covers all 200 Frank Merriwell tales. Avoid *premature formalization* by resisting the temptation to create numerous prototypes and attributes on speculation.

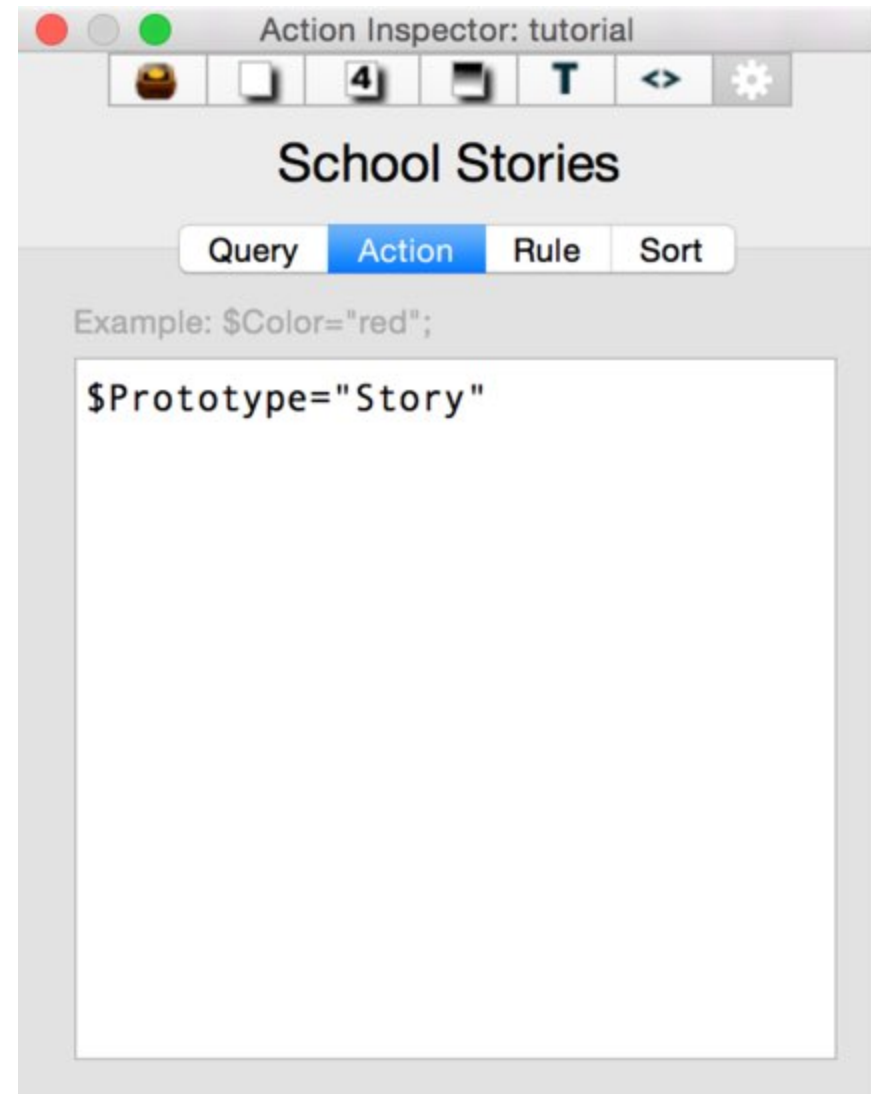
Begin with what you are confident you will need, and with what will clearly save time and typing. You can always add additional attributes later.

## 6. Actions

We now have a container named “School Stories” which holds roughly 18 notes. We have some films and some secondary sources and some other paraphernalia, but most of the notes describe stories. Looking ahead, we expect to add many additional notes to this container, and most of those notes will be additional stories.

It’s not very difficult to assign prototypes to newly-created notes, but we can ask Tinderbox to automate this by adding an **OnAdd action** to the School Stories container.

1. Select the Outline tab, and then select the container named “School Stories.”
2. Open the Inspector, and choose to the Action Inspector.



3. Select the Action pane if it's not already selected.
  4. Enter the OnAdd action **\$Prototype="Story"**:. Press [return].
- ☞ Note that Tinderbox actions are case-sensitive, and that rules use straight quotes ("), not typographic quotes (“”).

Whenever a note is added to the container School Stories, or is created inside that container, Tinderbox will now assign it the prototype **Story**. This action is only performed when the note is added or created, so you're free to change the prototype later if you wish.

- ☞ OnAdd rules frequently specify initial assumptions or guesses. Tinderbox, acting as your assistant, assumes new notes in this container should be Stories, but if you say otherwise, Tinderbox defers to your judgment.

## Composing Actions

Though Tinderbox actions can be both powerful and flexible, most rules are simple assignments:

```
$Color = "red"
$Width = 5.5
$Badge = "flag red"
```

Sometimes, an action needs to refer to another note. The action

```
$Width=$Width("Benjamin Franklin")
```



sets the width of *this note* to the width of the note named "Benjamin Franklin". If there might be several notes with the same name, we can specify the path to that note:

```
$Width=$Width("/scientists/Benjamin Franklin")
```

specifies the note named “Benjamin Franklin” inside the container named “scientists”. Tinderbox also provides some *designators* to refer to other notes:

```
$Width=$Width(parent)
```

sets the width of this note to be the width of the note’s container.

A series of actions may be separated by semicolons. The action

```
$Color="red"; $BorderColor="white"
```

sets both the note’s color and border color.

# 7. Sorting

## Adding some new notes

As our research progresses, we add additional notes.

Title: Autumn Term  
 Author: Antonia Forest  
 Year: 1948  
 Protagonist: female

Title: Ender's Game  
 Author: Orson Scott Card  
 Year: 1985  
 Protagonist: male

Title: Lord of the Flies  
 Author: William Golding  
 Year: 1959  
 Protagonist: male

Title: Intuition  
 Author: Allegra Goodman  
 Year: 2006  
 Protagonist: female

Title: The Magicians  
 Author: Lev Grossman  
 Year: 2009  
 Protagonist: male

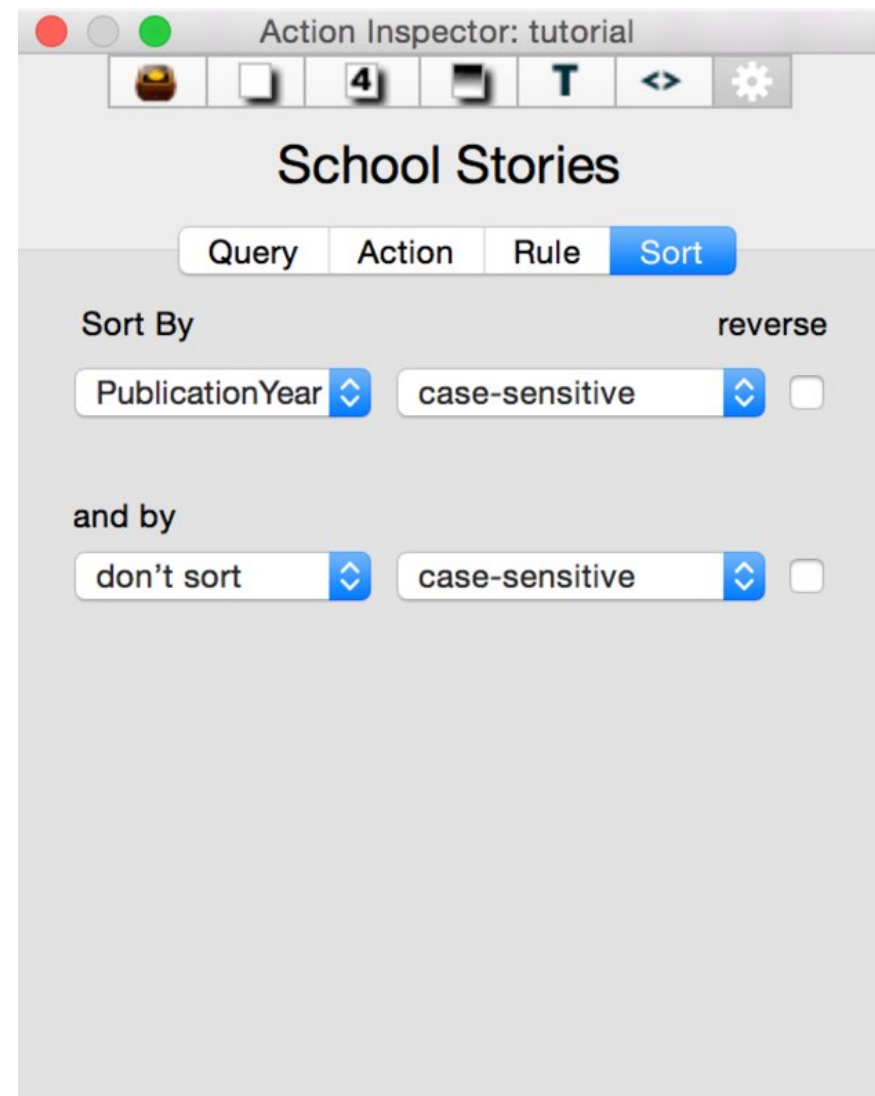
Title: Harry Potter (7 vols)  
 Author: J. K. Rowling  
 Year: 2006  
 Protagonist: male

At this point, also review each of the old notes and enter key attributes like Author and PublicationYear where this has not already been done.

## Sorting By Date

Select the tab for Outline View. Our notes are not in any particular sequence. That might be satisfactory in some cases. In others, we'll want to adjust the sequence manually. But often, lists can be made more manageable by instructing your Tinderbox container to sort them automatically.

1. Select the container "School Stories."
2. Open the Action Inspector.
3. Switch to the Sort pane
4. Select **PublicationYear**, which you will find in the References section of the attribute list, as the first Sort criterion.



The container of School Stories will now be sorted automatically by publication date. We begin with the oldest stories and proceed to more recent tales.

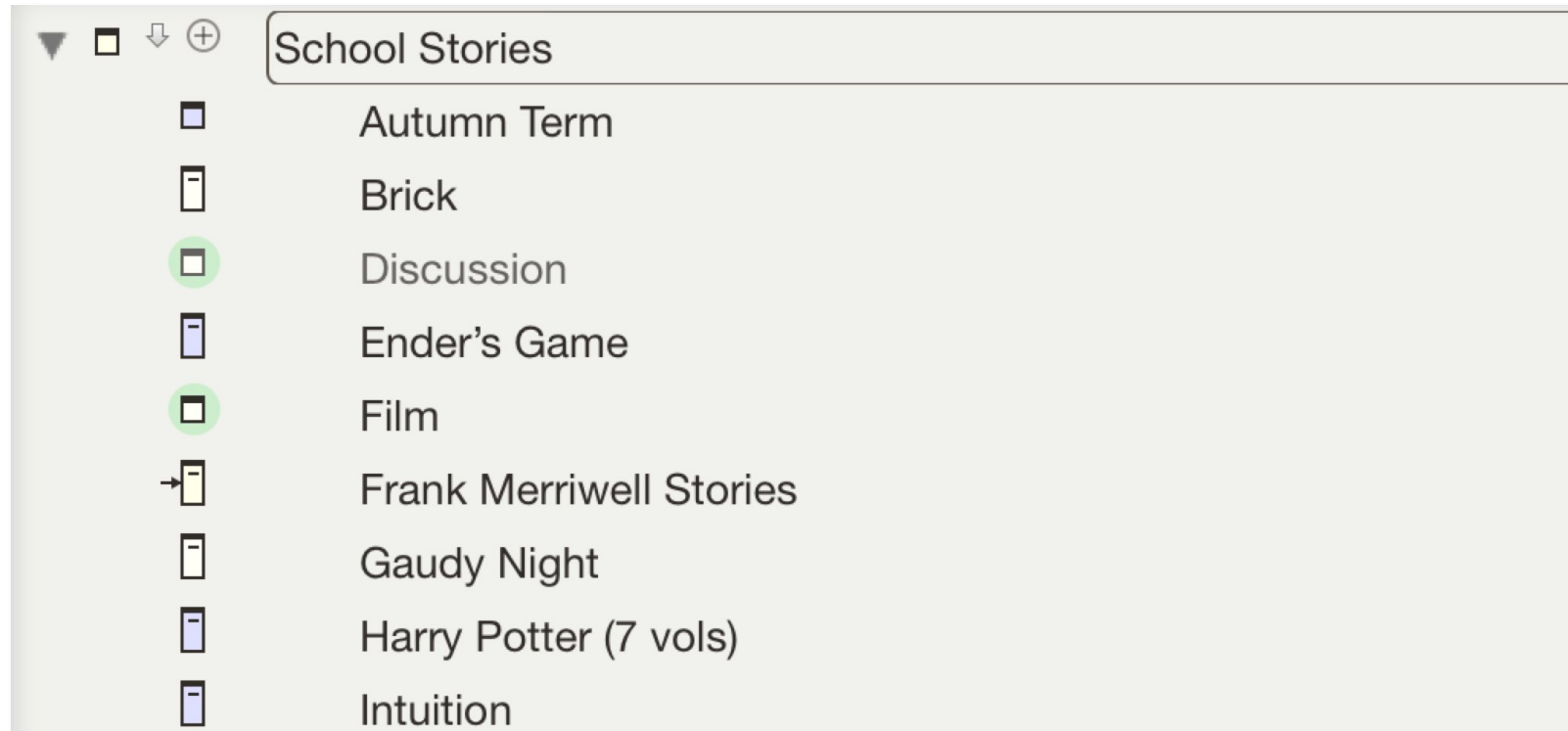


The sort inspector provides a checkbox that reverses the sort order, placing the newest titles first.

Note that sorting in outline view does not change the position of notes in the map view.

## Sorting by Name

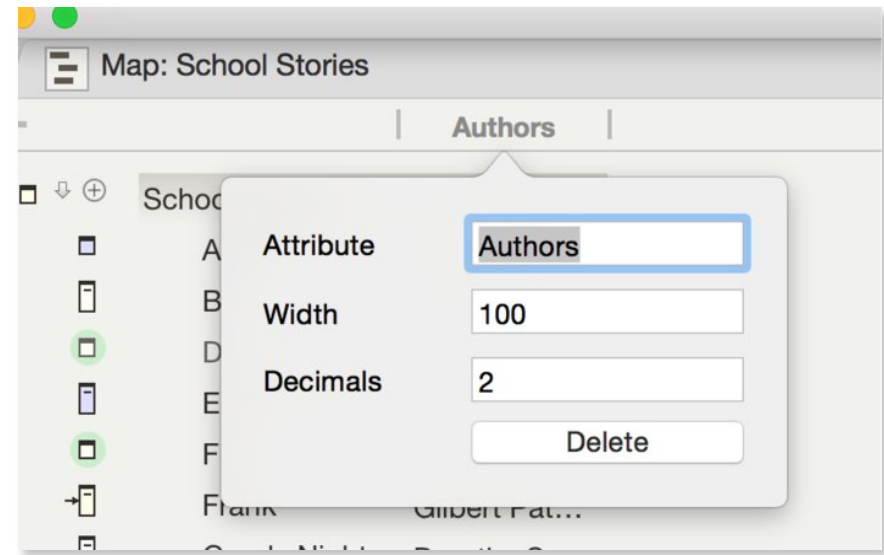
We might prefer to sort the notes alphabetically by title. Again, select the container School Stories and then use the Sort Inspector to select **Name** from the General section of attributes.



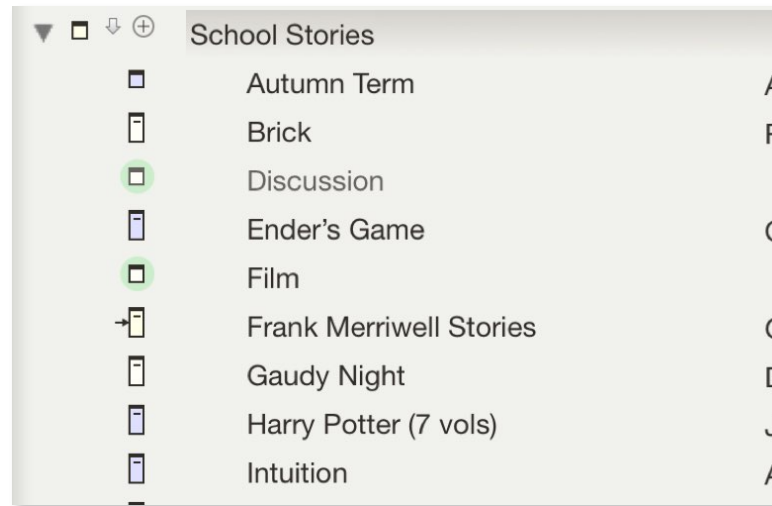
## Sorting By Author Name

Alternatively, we might wish to sort our stories by the author's name. To make this clearer, we'll first add a *column* to the outline to make the author's name visible.

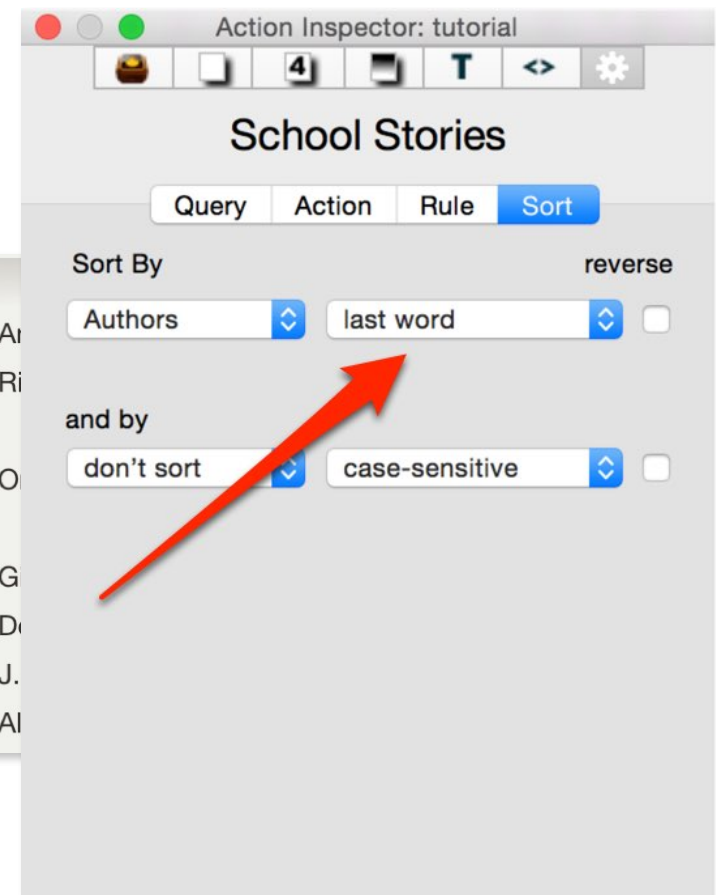
1. Switch to the Outline tab if you're using a different tab.
2. Select **Use Columns** from the View menu. A bar of column headings slides down from the top of the outline view.
3. At the left edge of the column bar, click on the “+” button to add a new column.
4. The new column is temporarily labeled “Attribute.” Click on the column label to change this to “Authors”, set the column width to 250 pixels, and then click outside the popover to dismiss it.



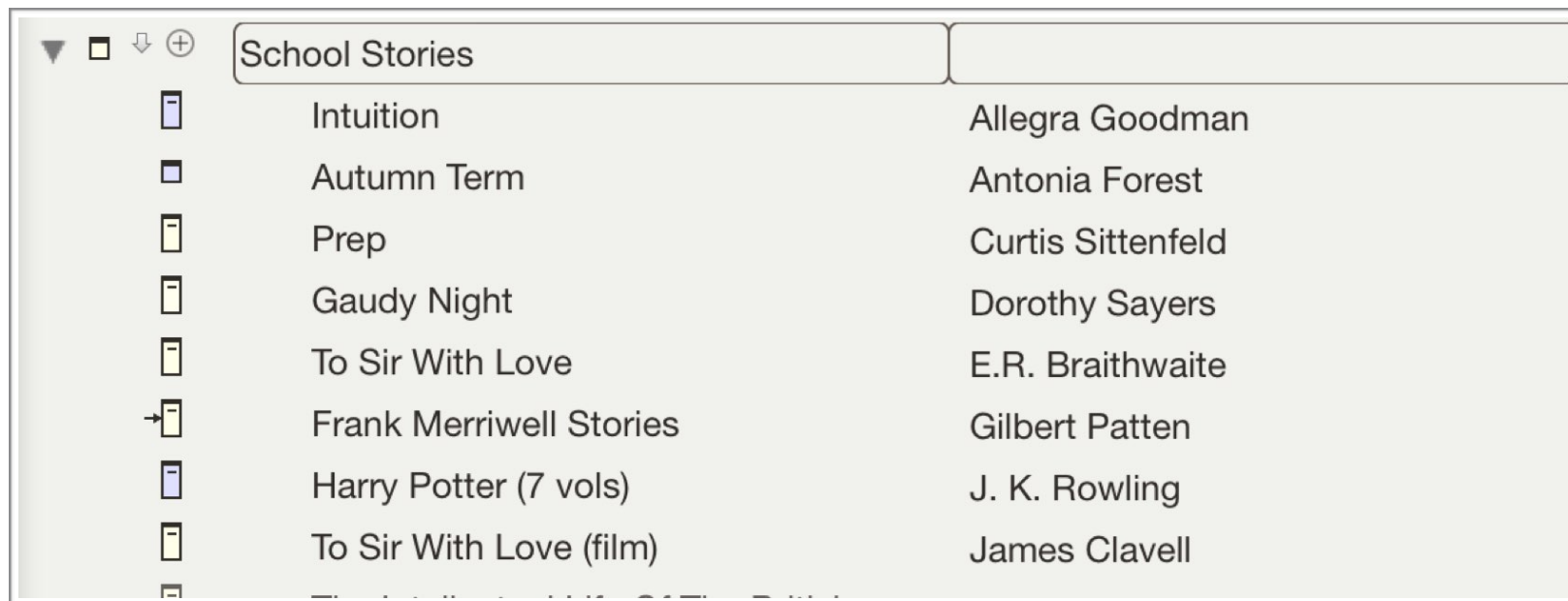
Adjust the column widths by dragging the column dividers; your outline – still sorted alphabetically by title, will look like this.











Now, open the sort inspector and, rather than sorting by Name, sort instead by



“Authors” in the Reference category of attributes.



School Stories		
	Intuition	Allegra Goodman
	Autumn Term	Antonia Forest
	Prep	Curtis Sittenfeld
	Gaudy Night	Dorothy Sayers
	To Sir With Love	E.R. Braithwaite
	Frank Merriwell Stories	Gilbert Patten
	Harry Potter (7 vols)	J. K. Rowling
	To Sir With Love (film)	James Clavell

The list is indeed sorted by the author name, but in this case we’re sorting by the author’s first name.

We might have anticipated this, of course, and listed author names in inverted order: “Goodman, Allegra” rather than “Allegra Goodman.”

Correcting this now might be mildly tedious, but it’s far from intolerable: we could easily correct several notes in a minute, and so a hundred notes might be a matter of an hour.

Instead, we’ll add a *sort modifier* in the sort inspector and ask Tinderbox to sort by the **last word** of the attribute’s value. This will give us the desired



sort order for most of our notes. It's not ideal: "Oliver Wendell Holmes, Jr." would be sorted under "J", which is wrong, and "Edward St. Aubyn" would be sorted under "A", with which some would agree and others would not.

Since these are working notes, this rough and ready solution may be perfectly adequate. Better solutions can in fact be achieved with agents, discussed below, if required.

## 8. The Attribute Browser

As we have seen, the Tinderbox Map view and the Outline view provide different ways of visualizing and organizing the same collection of notes. Map view places spatial organization and links in the foreground while viewing only one part of the hierarchy. Outline view places hierarchical organization in the foreground, while providing less scope for spatial representation and links.

Attribute Browser, a third Tinderbox view, explores the role of a specific attribute in your notes.

### Opening A New Attribute Browser

We'll create an attribute browser to examine the role of ProtagonistGender in our collection of school stories.

1. To begin, select the Outline tab if it's not currently selected.
2. Create a new tab by pressing the “+” button at the right edge of the tab bar. The new tab will initially contain a duplicate of the current view.
3. Choose **Attribute Browser** from the View menu.
4. From the popup menu labelled **Container**, select the container “/School Stories”.

- From the popup menus labelled Attribute, select the User category and then select the attribute ProtagonistGender.

The Attribute Browser begins by locating all the notes inside the designated container “School Stories”. Then, it examines the value of \$ProtagonistGender for each note. Three values are found to be in use: “female”, “male”, and “” (the empty string is the default value for \$ProtagonistGender, and is used in notes where we have not chosen a value for \$ProtagonistGender).

The screenshot shows the Attribute Browser interface. At the top, there are two dropdown menus: 'attribute' set to 'User' and 'container' set to '/School Stories/'. Below these are two columns: 'Name' and 'Authors'. The table is divided into two sections: 'female' and 'male'. Each section lists notes with a small icon to the left and the author's name in a blue box to the right.

female	
Autumn Term	Antonia Forest
Gaudy Night	Dorothy Sayers
Intuition	Allegra Goodman
Prep	Curtis Sittenfeld
Tam Lin	Pamela Dean
The Prime of Miss Jean Brodie	Muriel Spark
male	
Brick	Rian Johnson
Ender's Game	Orson Scott Card
Harry Potter (7 vols)	J. K. Rowling
Lord Of The Flies	William Golding
The Magicians	Lev Grossman
To Serve Them All My Days	R. F. Delderfield

- ☞ Notice that the Attribute Browser can use multiple columns, and in fact inherits the column setting from the outline view we were viewing when we switched to Attribute Browser.

## Organizing Notes in The Attribute Browser

When you drag notes to a different section of the Attribute Browser, Tinderbox changes their value to match the new location. For example, if we

drag the notes “Frank Merriwell Stories” into the section for which \$ProtagonistGender is “male”, then Tinderbox will automatically assign that value to the note you just dragged.

## Summaries in Attribute Browser

Click the **disclosure triangle** at the upper left-hand corner of the Attribute Browser’s toolbar to reveal additional options.

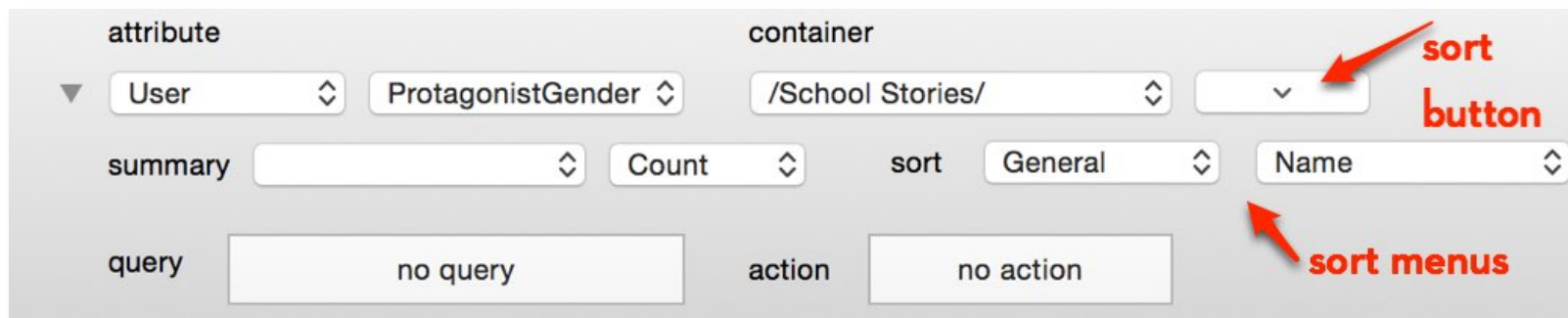
First, you can add a count of the number of stories in each section. From the first **summary** popup, select Name. From the second summary popup, select Count. At the right edge of each section header, Tinderbox displays the number of notes in each category. At present, we have seven female protagonists and a dozen male protagonists.

The first summary popup lets you choose to summarize any column of the attribute browser. If the column is a string like Name or Authors, the summary can only provide a Count. If the column is a numeric attribute, the summary can display the total, the average value, the largest value, or the minimum.

☞ In Attribute Browser, edit columns by right-clicking or control-clicking the column headers.

## Sorting in Attribute Browser

The **sort button** lets you reverse the sort order of Attribute Browser categories.



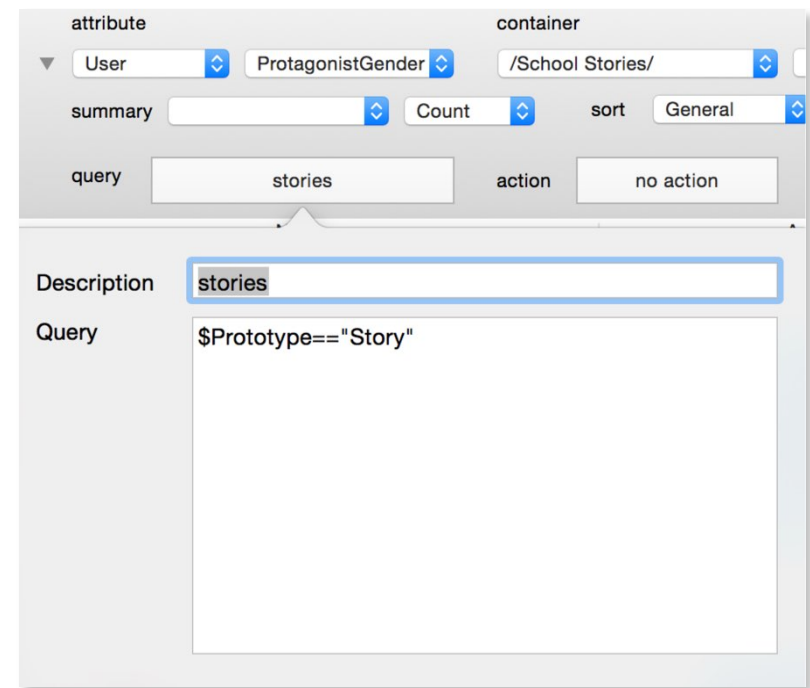
The **sort menus** let you sort items in Attribute Browser within categories. Here, we ask Tinderbox to list notes alphabetically by name.

## Restricting Our Scope

Let's turn our attention to the third category in Attribute Browser: the notes which have no value for ProtagonistGender. These might include several different kinds of notes.

- Notes for which we've simply neglected to specify a ProtagonistGender. This is a good opportunity to correct such oversights by dragging the notes into the “male” or “female” lists.
- Notes for which the choice of gender is unclear. (It could be argued, for example, that *Intuition* has two protagonists, one of whom is male.)
- Notes for which the choice makes no sense – secondary sources, prototypes, and other notes that don't represent school stories.

We can concentrate our attention on notes for which this attribute browser is pertinent by adding a *query*. The query is a test that will be applied to each note in the target container; only notes that satisfy the query by passing the test will be included in the final view.



1. Press the Query button to display the query popover.

2. Enter the query **\$Prototype=="Story"**, which will be true for notes which use Story for a prototype and false for other notes. (The “==” operator tests for equality; queries are explained in detail below.)
3. Enter a description of the query for future reference
4. Click outside the popover to dismiss it.

These queries excludes non-fiction discussions of school stories, since their prototype is “Discussion,” and also excludes notes about movies, since their prototype is “Film.”

Two notes remain with no \$ProtagonistGender – the prototypes Discussion and Film, each of which uses the prototype “Story” and so satisfied the query. We can simply ignore these, or we could exclude them from view with a slightly more elaborate query:

**\$Prototype=="Story" & \$IsPrototype==false**

This query is satisfied only by notes that use Story as a prototype *and* which are not themselves prototypes.

- 👉 In Tinderbox, “==” is the *equality operator* and tests whether two things are equal, while “=” is the *assignment operator* and assigns the value of an attribute.

## 9. Aliases

A Tinderbox *alias* is a special kind of note that stands for another note, just as a Macintosh *alias* is a file that stands for another file. A note can have many aliases; each alias corresponds to one specific note – its *original* note.

Most properties of an alias are taken from the original note. For example, if the original note is blue, its alias will be blue. If the note's color is changed to black, the alias will also change to black. Changes flow in both directions: if you change the color of the alias to red, the original (and all other aliases) will also be red.

☞ A few attributes are *intrinsic* and aren't forwarded along by aliases. For example, each alias has its own map position attributes \$Xpos and \$Ypos.

Aliases are permanently associated with their original note. If the original is deleted, all its aliases will be deleted. Deleting an alias has no effect on the original.

To create an alias:

1. Select a note.
2. Choose **Make Alias** from the Edit menu.



Aliases are indispensable when a single note needs to be in two different places. For example, in a list of Presidents of the United States, Grover Cleveland would be both the 22nd and the 24th president. In a faculty directory, the same professor might sometimes hold a position in two different departments. In a list of tasks, the same task might be listed under “important tasks,” “things to be done in the office,” and “things that must wait until Martha is available.”

Aliases are important to Tinderbox agents, discussed below.

Aliases are drawn like regular notes, but the title of an alias is conventionally drawn in italics. Because some languages don’t use italics, you can request (in Document Settings: Outlines) that Tinderbox also underline alias titles.

## 10. Agents

Tinderbox agents constantly scan your document, looking for notes that match a specified criterion. For example, one agent might scan for Overdue Tasks, another might look for any notes that mention “Muriel Spark,” and a third might look for stories published before 1938.

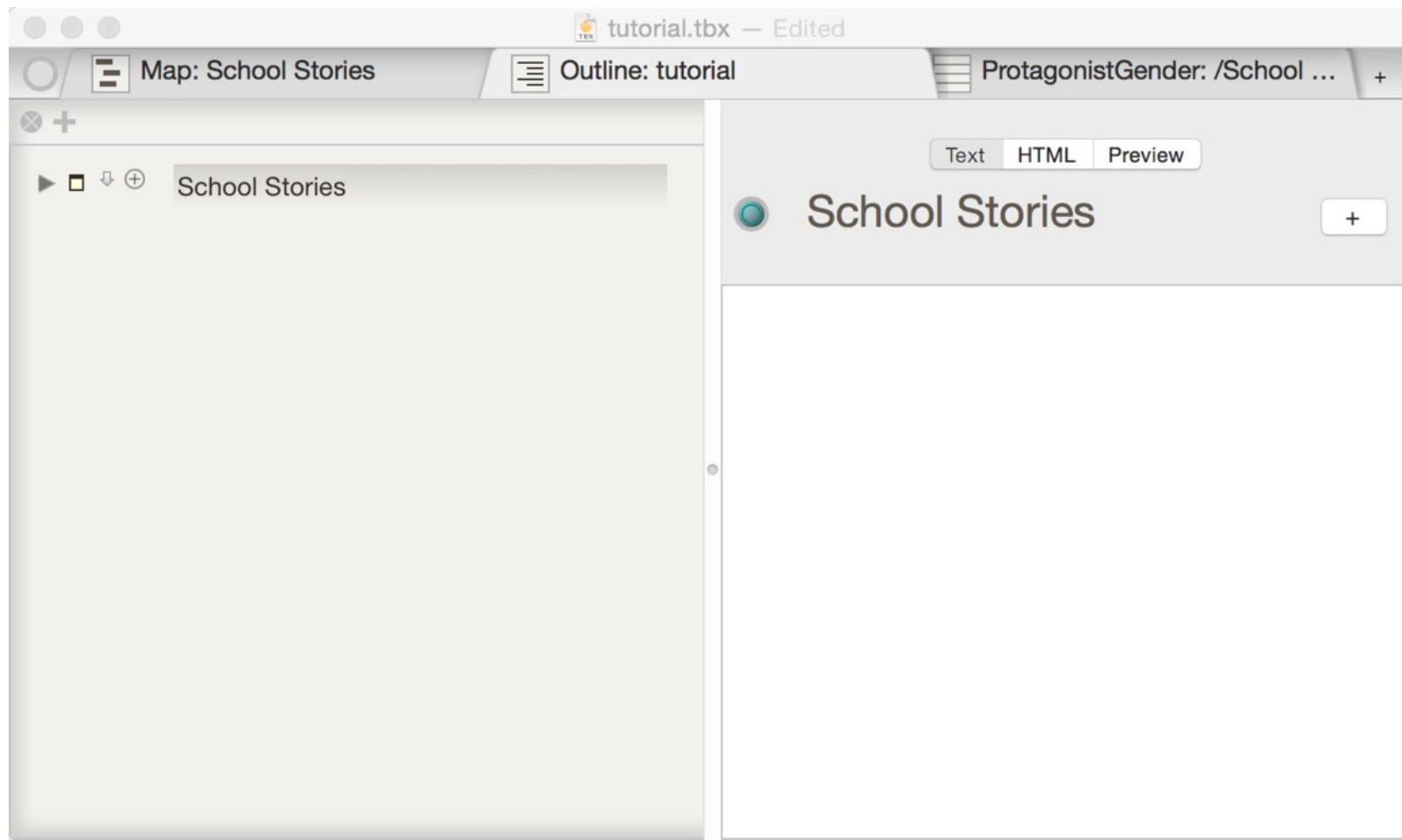
Agents are a powerful and distinctive feature of Tinderbox, helping documents organize themselves autonomously but not interfering with your work. Yet not every project requires agents, and many require only simple agents. Similarly, it’s seldom important to design all your agents before you begin to make notes, and most people find it easier to add agents gradually as the work proceeds.

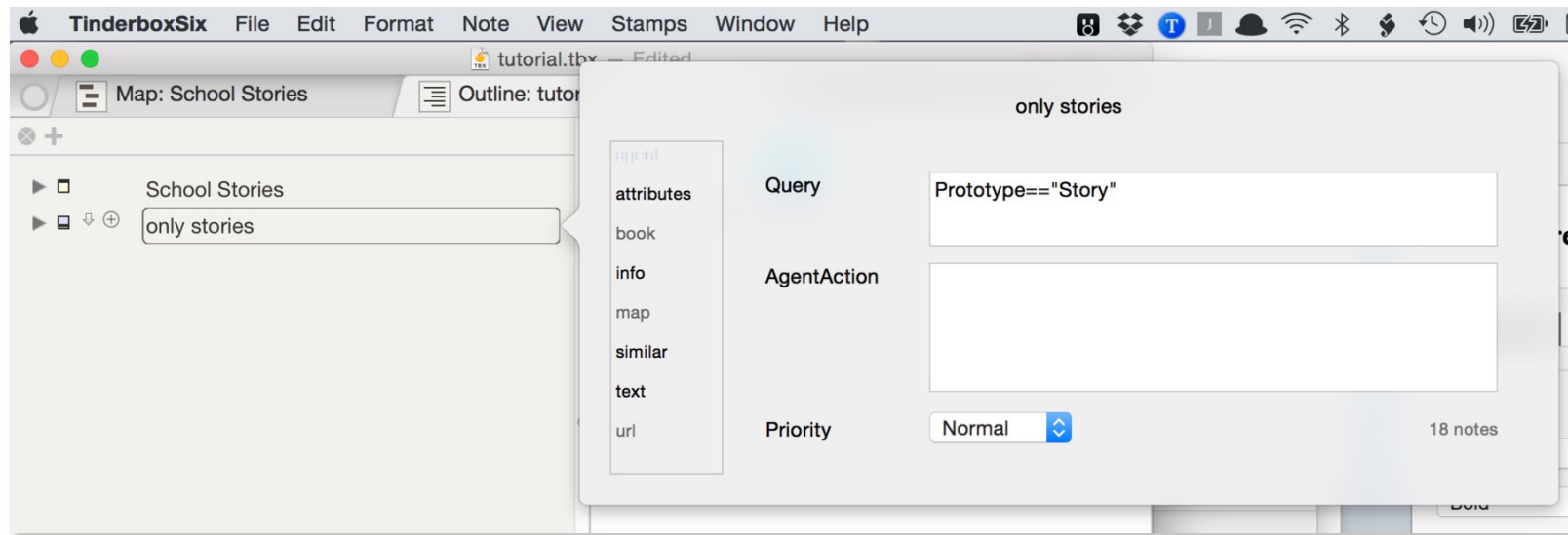
While some agents may prove helpful throughout the course of a project and indeed may help keep a large project properly organized, other agents may be improvised to perform a specific task or to answer a specific question. New agents can be improvised when you need them and, their work done, may be deleted when no longer needed.

# Making An Agent

Let's begin by making an agent that collects all the school stories in our collection, ignoring films, secondary sources, and any other kinds of notes.

1. Select the tab for Outline view
2. If the breadcrumb bar is present, click on the leftmost tab to move to the top level of the document. The breadcrumb bar will close automatically.
3. Click the expand/collapse triangle of **School Stories** to collapse it.





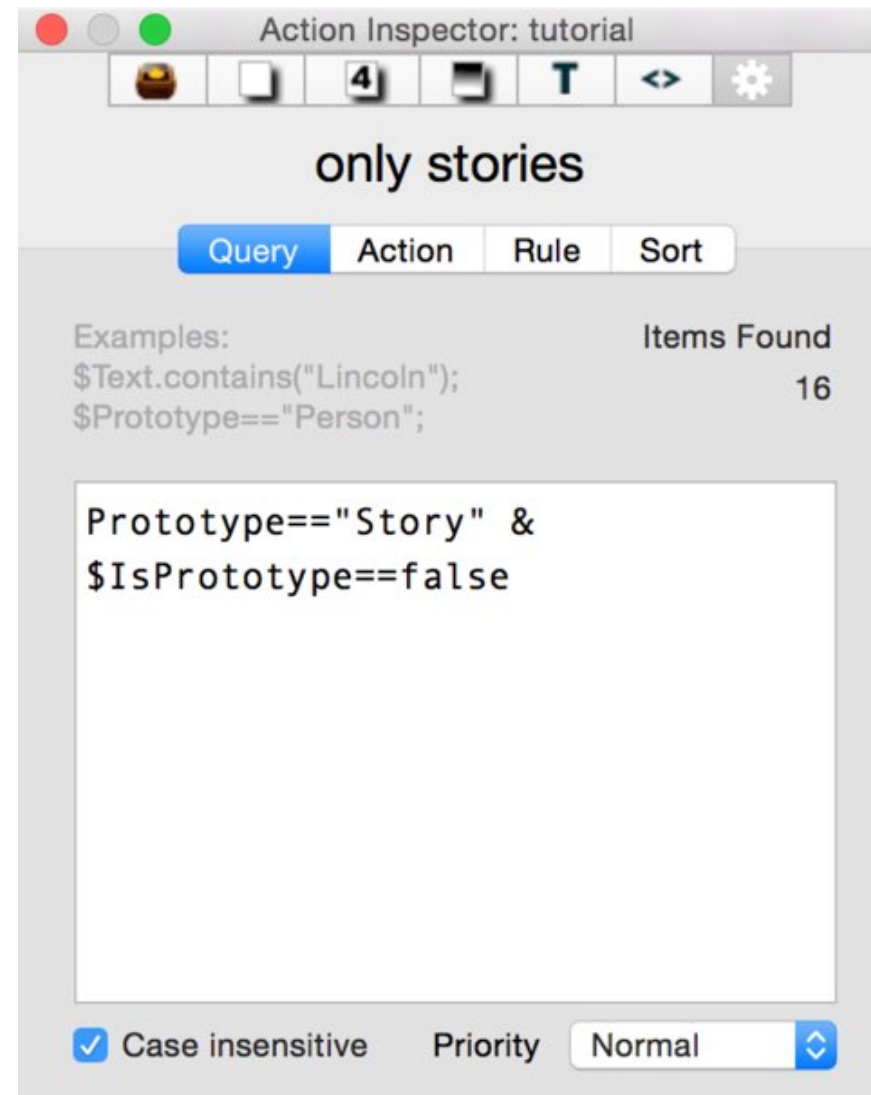
4. Choose **Create Agent** from the Note menu. A new agent will be created beneath School Stories.
5. The new agent is initially untitled. Name it “Only Stories,” and press [return] to complete the agent.
6. The Get Info popover will appear automatically to let you tell Tinderbox what you’d like this agent to do. You can display this popover whenever you like by selecting an agent and choosing **Get Info** from the Note menu and, if necessary, switching to the “agent” pane.
7. The agent’s query describes what sort of note the agent looks for. In this case, the query is **Prototype=="Story"**. This will be **true** for notes that use the Story prototype, but **false** for notes that use Discussion or Film, or for notes with no prototype.
8. Click outside the popover to dismiss it.

9. The agent now has children, and a disclosure triangle; click the triangle to view the notes the agent has gathered.

In this case, we see that the agent has gathered notes for our books, excluding films and discussion. As was the case in the Attribute Browser, we've also collected aliases for the prototypes Film and Discussion, since these also use **Story** as a prototype.

The solution is exactly the same as our solution to the Attribute Browser issue: we change the query. This time, rather than use the Inspector we'll explore the **Get Info** popover, a tool that can provide many insights into our notes.

1. Select the agent "only stories."
2. Choose **Get Info** from the Note menu.
3. The agent pane may already be selected. If not, choose it.
4. Change the query to **\$Prototype=="Story" &**



`$IsPrototype==false`, and press [return].

5. Click outside the popover to dismiss it.

## The Agent Inspector

You can use the Inspector to examine agents and to change their properties. Simply select the agent, open the Inspector, and switch to the Action Inspector (the right-most inspector tab).

The **Query** pane lets us change the query, much as the Get Info agent pane does. We can also use the action inspector's **Sort** pane to change the sort order for an agent.

1. Select the “only stories” agent.
2. Open the Action Inspector and switch to the Sort pane.
3. Sort by **Name** (in the General category of attributes).

The aliases inside our agent are now sorted alphabetically, beginning with *Autumn Term* and ending with *Wonder Boys*.

## Agent Actions

An agent can perform one or more *actions* on the notes it gathers. Even simple actions can be very helpful. Actions can automatically tie visible properties of notes to their internal structure, can move notes around in your document, can help ensure that the notes are internally consistent, and can call your attention to notes that are particularly interesting, urgent, or that require your attention.

Here, we'll create an agent that locates stories that have a female protagonist, and set those stories to display their title in bold type.

1. Select the “only stories” agent, and collapse the agent if it is expanded.
2. Create a new agent by choosing Create Agent from the Note menu. Name the new agent “female protagonists.”
3. The query for this agent is **inside("/only stories") & \$ProtagonistGender=="female"**. Note how this agent uses the work already done by our first agent to examine only notes that are stories.
4. The **action** for this agent is **\$NameBold=true**. (Note that the Boolean values **true** and **false** are keywords and require no quotation marks. **true** is a boolean value; "true" is a string with four letters.)
5. After entering the action, press [return] and click outside the popover to dismiss it.

6. Expand the newly-created agent to examine what it has found.

The new agent has located several stories with female protagonists: *Autumn Term*, *Gaudy Night*, *Tam Lin*, and others.

Collapse the agent for a moment and expand School Stories. Notice that stories with female protagonists are displayed in bold type, while other notes are not.

## Watching Agent Progress

Agents are quite fast, and Tinderbox Six usually performs agent chores in the background so you won't need to wait. In documents that have thousands of notes and numerous complicated agents, it's possible that your agents might take a few seconds to catch up with changes in your document.

The **Agents & Rules** pane of the Tinderbox Inspector shows you an animated view of the progress of agent updates. A vertical blue bar indicates what part of your document has been most recently scanned, and the bar also indicates which agent is currently active. In this small document with only two simple agents, updates will be much too fast to notice, but the Agents & Rules inspector can be handy when documents grow complex.



Occasionally, you might prefer to turn off automatic agent updates entirely. If so, choose **File ▶ Update Agents Automatically** to turn off automatic updates or to re-enable them. When automatic updates are disabled, you can choose **File ▶ Update Agents Now** to update whenever you like.

## Automatically Extracting Values

The attribute \$PublicationYear is a string. Tinderbox dates always include a specific calendar date and time, and for most purposes we only need (or know) the year of publication. At times, though, we may want to have a date attribute corresponding to \$PublicationYear for internal calculation.

We could, of course, simply go through each note and type a value for \$StartDate. If a book was published in 1974, we'd enter (say) 1/1/1974. That's easy enough for a few dozen notes, but we can have an agent do the work for us.

First, we'll create an agent that gathers all notes inside School Stories that have a PublicationYear. Make a new agent and name it “date agent”. Its query is:

```
inside("/School Stories") & $PublicationYear!=""
```

☞ In Tinderbox queries and actions, the empty string is written as two adjacent quotation marks. The operator `!=` means "not equal", and is the opposite of the operator `==`.

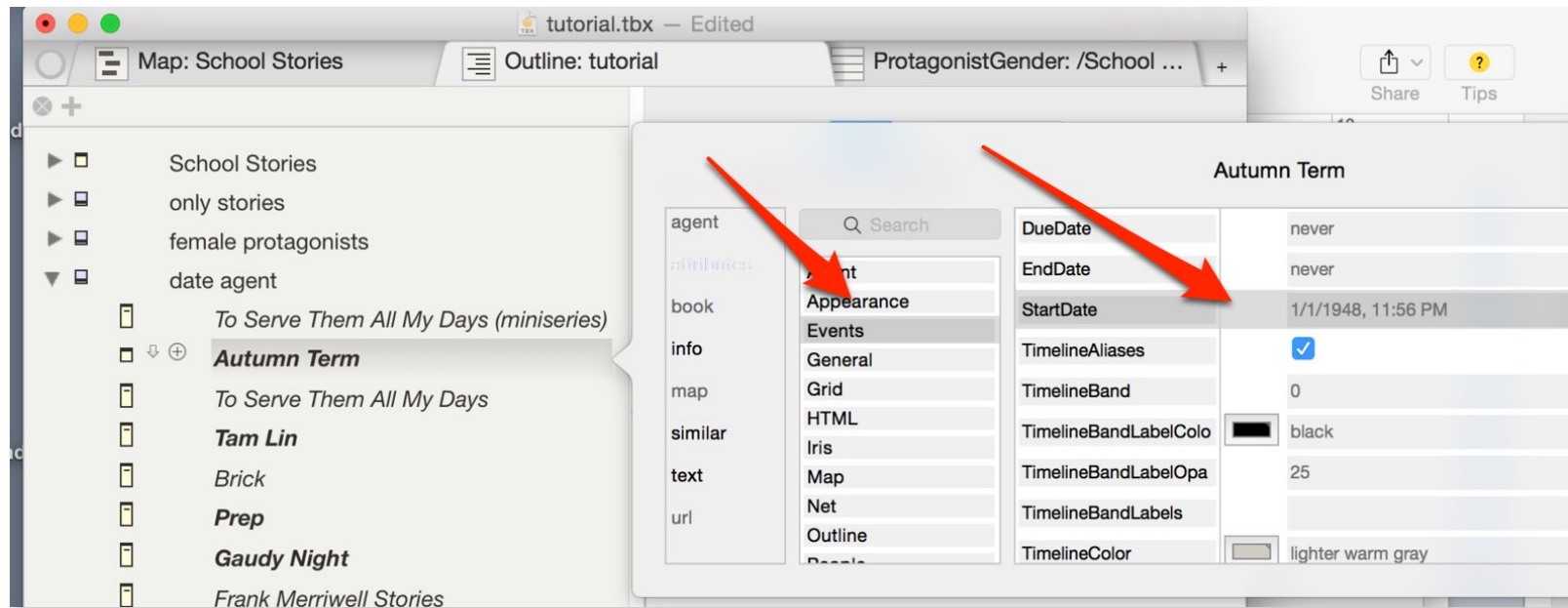
The action for this agent is

```
$StartDate.year=$PublicationYear;
$StartDate.month=1;
$StartDate.day=1;
```

Several distinct actions may be strung together, separated by semicolons. Line breaks are added here for clarity; in the popover or inspector, simply type the action in one line.

Note that this is only one of several ways we can set the date. Tinderbox provides a varied repertoire of functions for working with dates, and throughout Tinderbox (as we have seen) there are often several different ways to accomplish the same task.

To check that the agent is doing what we expect, select one of the notes inside the new **date agent** – perhaps *Autumn Term*. Choose **Get Info** from the Note menu, and select the **attributes** pane of the get Info popover. The popover will display a list of Attribute families; select Events to view a list of attributes relating to events. One of these attributes is `$StartDate`; the agent should have set the `$StartDate` to 1 January 1948.



If you aren't sure where to find an attribute, the Search tool in the Get Info popover will help find it.

## Agent Queries

Each agent has a *query*, stored in `$AgentQuery`, that describes the notes for which it is looking. Each note in the document is examined in turn; if the note satisfies the query its alias will be placed inside the agent.

Queries are *expressions* that evaluate to **true** or **false**. The details of actions and queries will be found in Tinderbox Help, but most queries are quite simple. Some examples may be instructive.

---

Stories added this week

```
$Prototype=="Story" & $Created<date("today-7 days")
```

---

Notes modified today

```
inside("/School Stories") & $Modified>date("yesterday")
```

---

Stories that have any kind of badge

```
inside("/only stories") & $Badge!=""
```

---

All notes which have a specific badge

```
$Badge=="question"
```

---

All work published before 1945

```
$StartDate!= 0 & $StartDate<date(1,1,1946)
```

---

Films for which you don't know the release date

```
$Prototype=="Film" & $StartDate== 0
```

# 11. Table Expressions

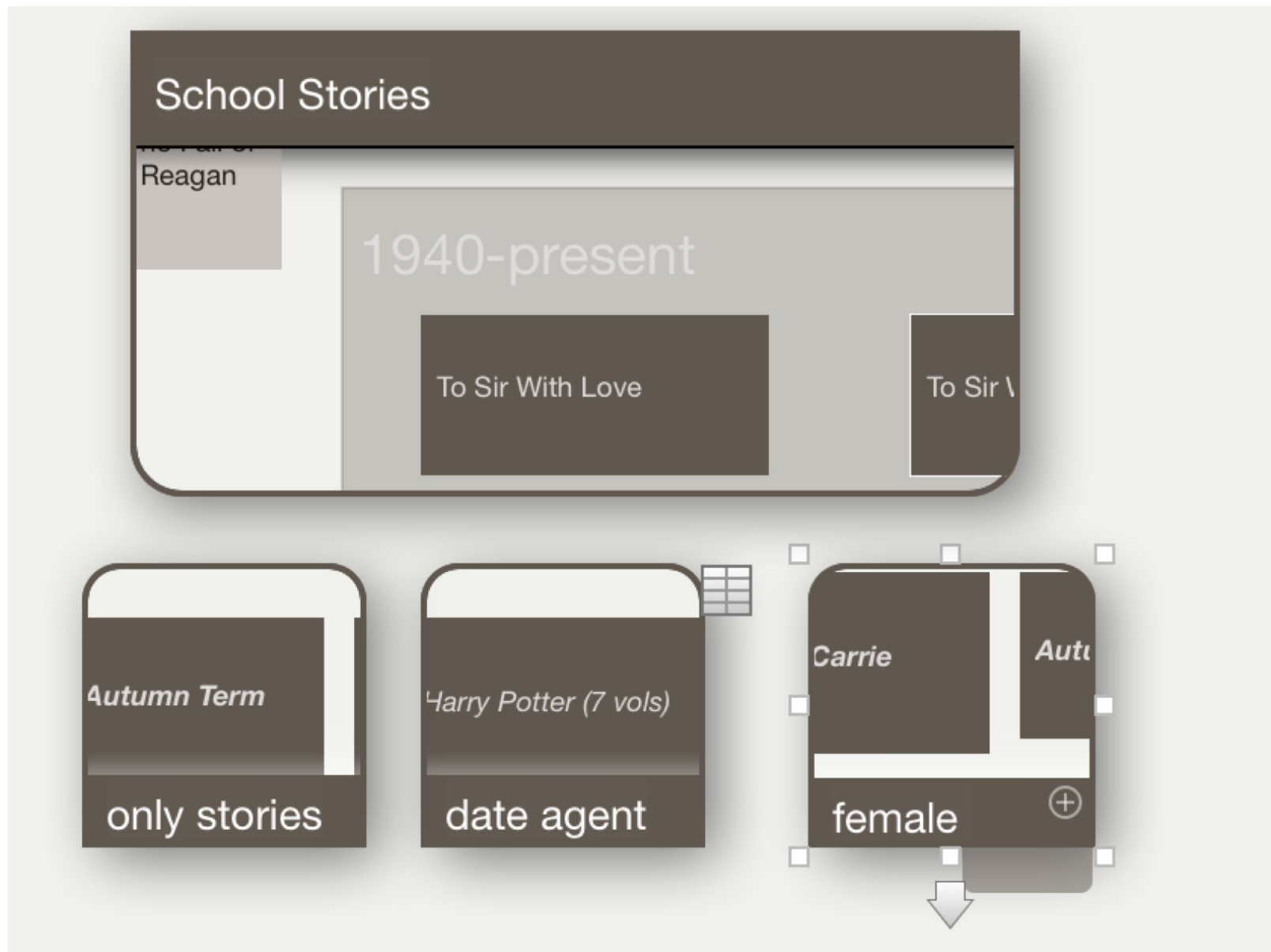
## Agents and Containers

We've been working with agents in outline view. Let's examine them in map view as well.

1. In Outline View, if the breadcrumb bar is visible, click on the leftmost breadcrumb "tutorial" to view the top level of the document.
2. Create a new tab by pressing the "+" button at the right edge of the tab bar.
3. Choose **Map View** from the View menu.

The arrangement of notes in the map will be arbitrary, as you've not previously placed the new agents in a map view. Drag them to convenient positions, or choose **Cleanup ▶ To Grid** from the View menu to arrange the notes automatically.

Containers like **School Stories** have a title bar at the top of the container. Agents, such as **only stories** and **date agent**, have a title bar at the bottom of the agent. This lets you distinguish containers and agents at a glance. In addition to the title bar, agents and containers have a *viewport* that lets you see some of their contents.

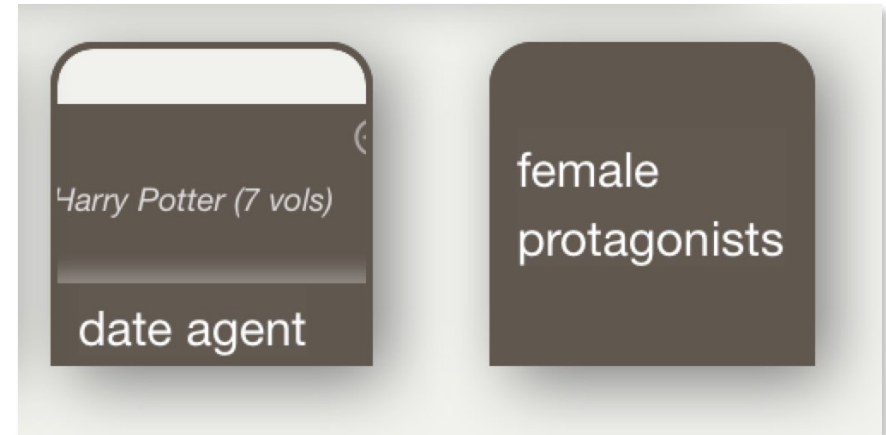


- ☞ To *zoom into* an agent or container, select it and press down-arrow.
- ☞ To *zoom out* of agent or container, use the breadcrumb bar or press up-arrow.

If you like, you can see more notes inside an agent or container by dragging its resize handles to make it larger.

## Adjusting The Title Bar

The title bar of a new note or agent might not provide sufficient space to display the entire title. If you like, you may increase the size of the title bar and decrease the viewport.



1. Place the mouse over the bottom of the title bar of a container, or the top of the title bar of an agent — the boundary between the title bar and the viewport. The cursor will change to a resize cursor.
2. Click and drag the title bar up or down.

For this exercise, move the title bar of the new agent **female protagonists** so it fills the entire agent.

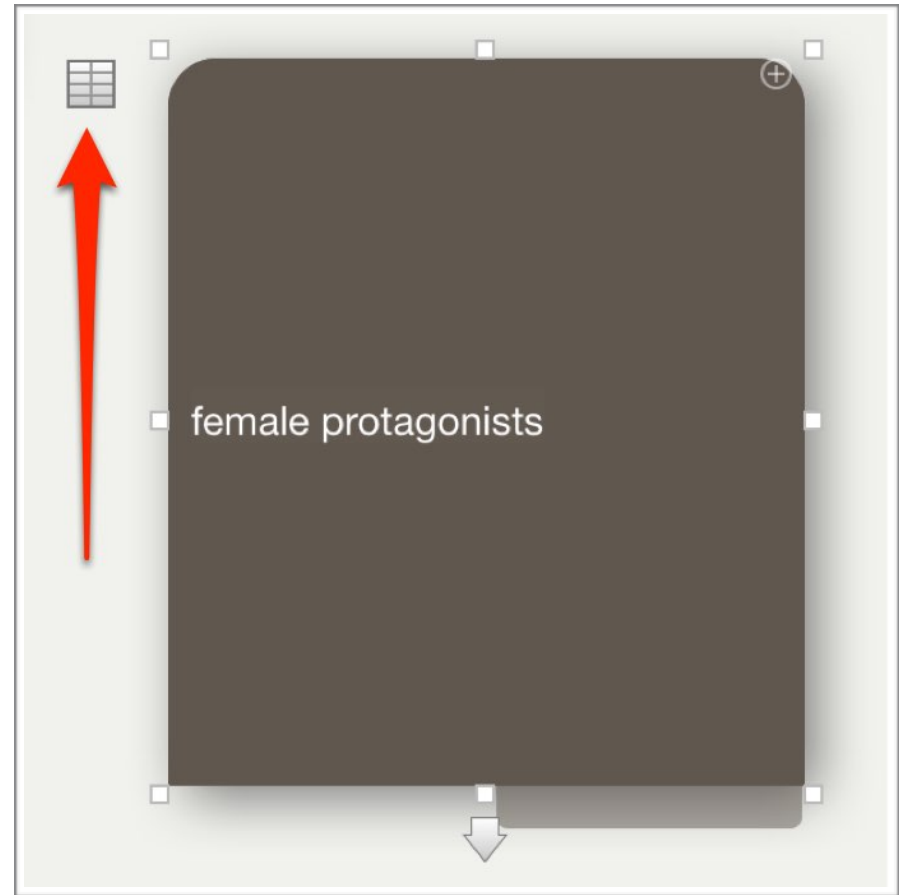
## Adding A Table Expression

Containers and agents may present a *summary table* in their title bar to summarize the notes they contain. The table is only displayed in the title bar

has sufficient space; if there is not sufficient space to display the entire table, only the first few lines will be shown.

To add a table to the agent **female protagonists**:

1. Drag **female protagonists** aside so there's plenty of empty space around it, and increase its width and height.
2. Resize the title bar to completely fill the agent
3. Click on the table widget to display a popover.
4. Click the “+” button to add a new column to the summary table.
5. Select the new line and press [return] to edit it. Under “Heading,” enter the heading “title”. Tab to the Expression column, and enter “\$Name”.
6. Click outside the popover to dismiss it.

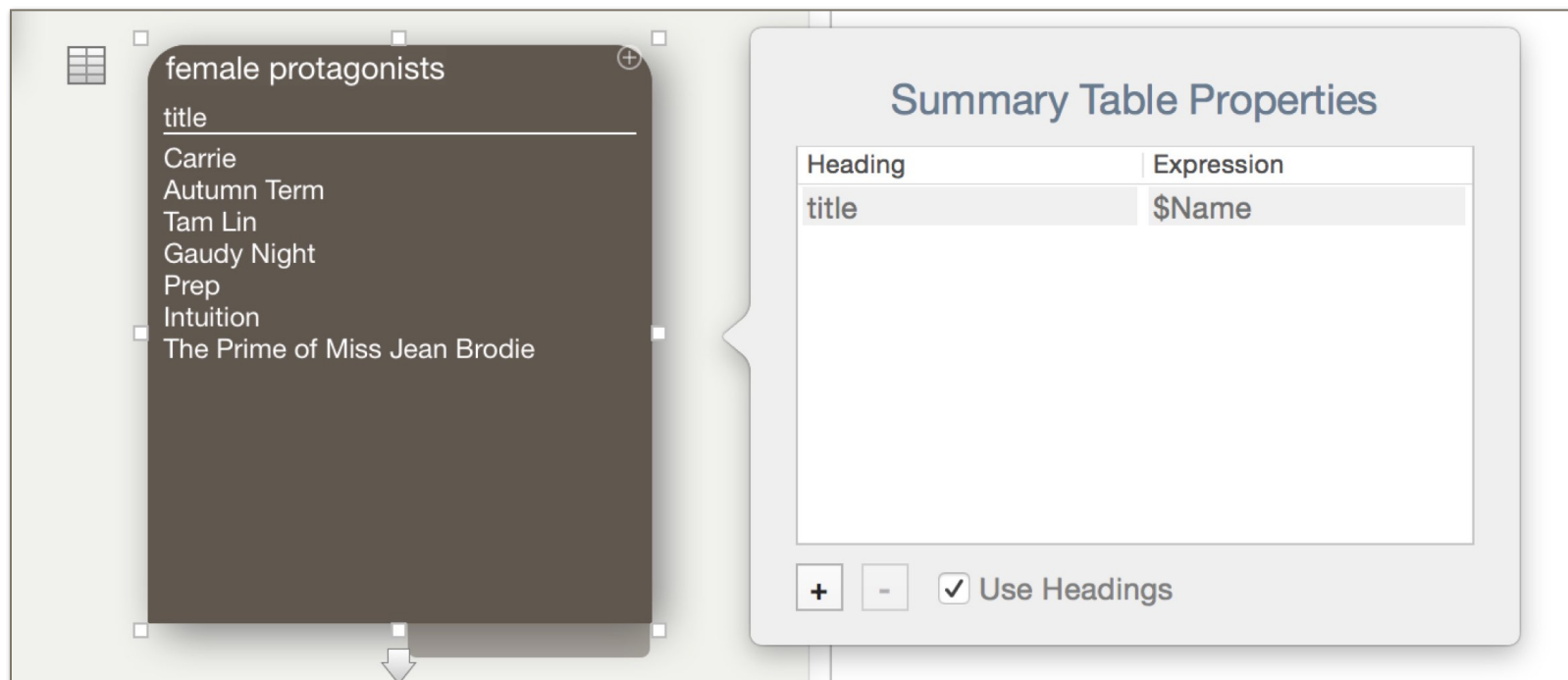
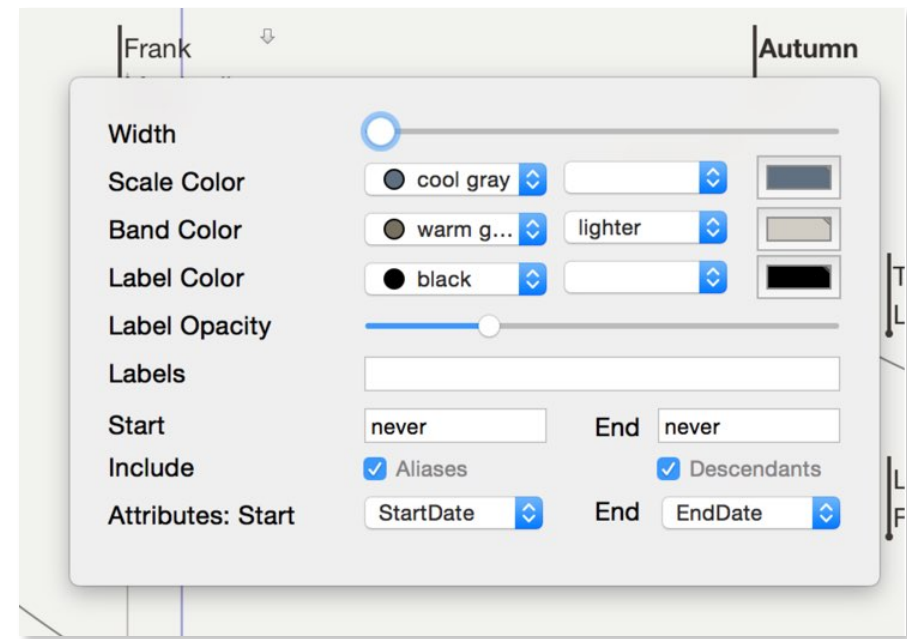


Space permitting, each child of the agent or container will be represented by one line of the summary table. The order of items in the summary table is



the same as the order of items inside the container or agent, so sorting the container will also sort of summary table.

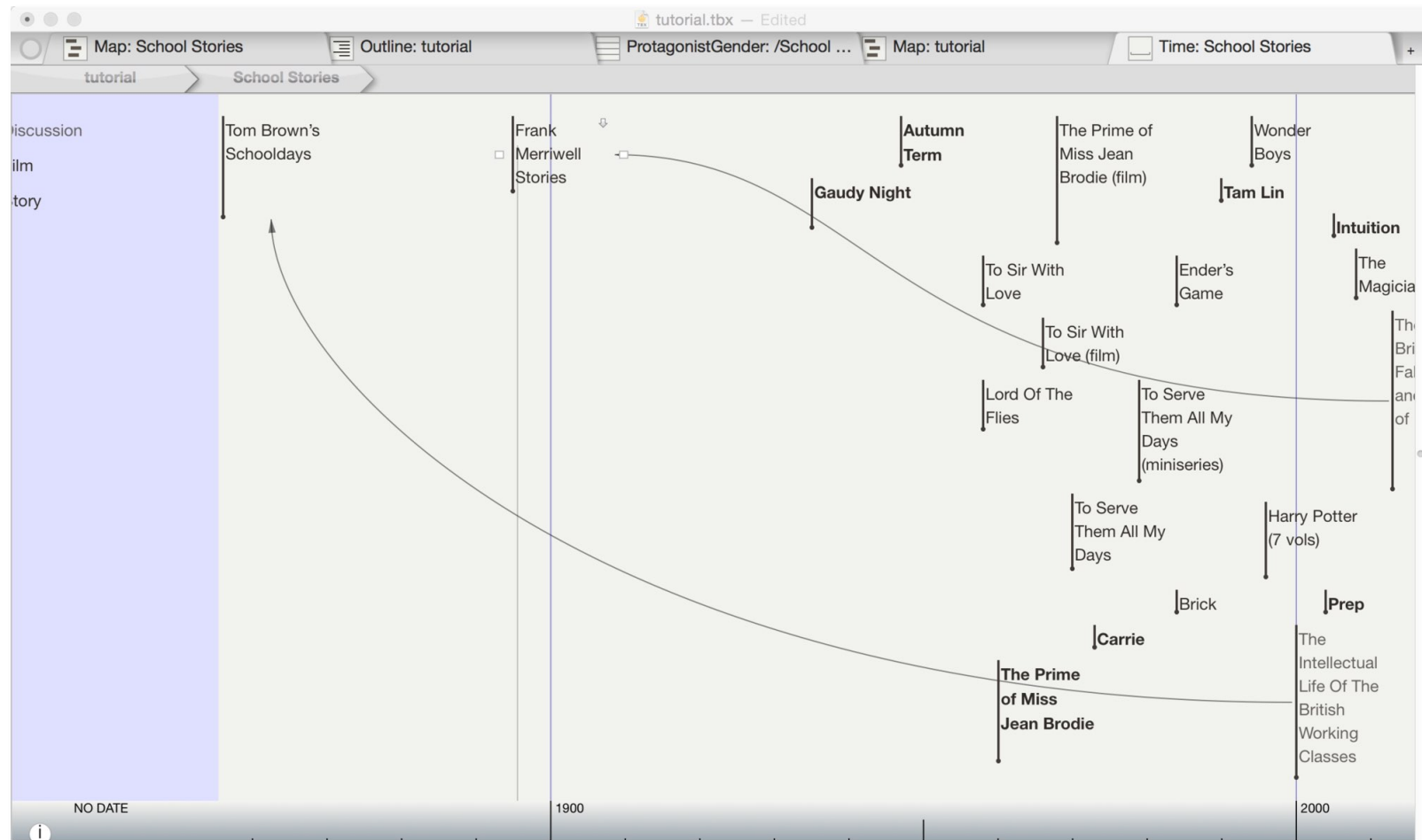
Additional columns may be added to the summary table by opening the Summary Table Properties popover and pressing + to add additional columns.





# 12. Timeline

The Timeline View can be useful for exploring a sequence of events. Tinderbox makes it easy to explore a timeline of notes in any agent or container. Let's begin by taking a look at everything inside our container of School Stories.



1. Switch to the tab for the Map View of School Stories.
2. Create a new tab for our timeline.
3. Choose **Timeline** from the View menu.
4. Since the timeline takes up a good deal of space, slide the pane splitter to the right to use most of all of the window for the timeline view.

The timeline view displays each inside the School Stories container. Those that have no date – the prototypes Story, Film, and Discussion – are listed at the left. All other notes are shown in a temporal sequence, ordered by the value of \$StartDate.

## Timeline Settings

A variety of timeline parameters and settings can be adjusted by clicking on the Info button at the left edge of the timeline view's time scale, at the bottom of the window. Here, you can increase or decrease the width of the timeline, change the background color, or change the timeline's start and end dates. Move the Width slider to its midpoint to give the timeline some breathing room. Click outside the popover to dismiss it.

- ☞ By default, the start and end dates are based on the earliest and latest events in your notes. You might wish to choose an even later end date to allow a little extra horizontal space for note titles.

## Timeline Bands

Often, timelines are clearer if we separate different kinds of notes into separate timeline *bands*. In some timelines, different bands might represent different places: the Kings of England might appear in one band and the Kings of France in a separate band. Or, different bands might represent different sorts of events entirely: a band for political events and another for cultural landmarks.

The band in which a note appears is determined by the attribute `$TimelineBand`. You can move a note to a new band by dragging it vertically, and a note's date can be changed by dragging it horizontally. (Note that we have an agent that automatically sets `$StartDate` from the note's `$PublicationYear` attribute, so if you move a note horizontally, it will snap back to its position when the agent runs!)

1. Select **The Prime of Miss Jean Brodie (film)**.
2. Open the Properties Inspector and select the Quickstamp pane.

3. Choose the Events family from the first popup menu.
4. Select the attribute TimelineBand from the second popup menu.
5. The current value of \$TimelineBand is 0. Change it to 1, and press [return].

The selected note will move to a band beneath the current timeline. Now, select **To Sir With Love (film)**, and assign it to \$TimelineBand 1 as well. While you're at it, move the two Discussions to TimelineBand 2. Your new timeline should be somewhat better organized.

## 13. Brainstorming With The Map

The map view is usually best used in the way we have already used it – to cluster related notes and concepts. In our early work, we designated one part of the map view of School Stories to hold early stories, another part of the map view for later work, and a separate part was reserved for discussions of school stories.

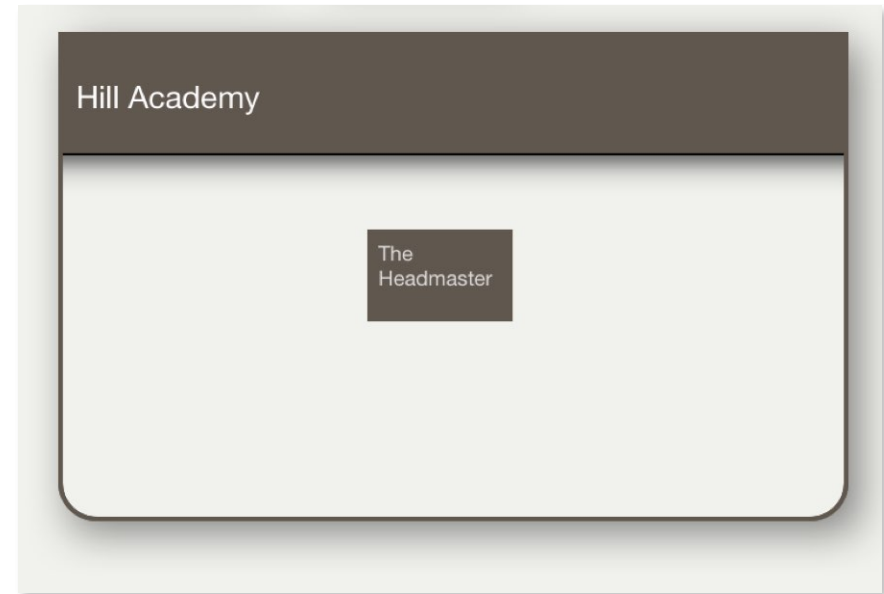
We may also use the map view more literally to represent spatial relationships. Here, we combine the abstract and literal approaches to develop ideas about our own school story, which will be set in a fictitious school called Hill Academy. In this exercise, we'll quickly jot down some ideas for the school's buildings and inhabitants, gather some visual references, and experiment with some ways to describe relationships.

### Sketching A World

To begin, we'll make a new container that will hold our school map.

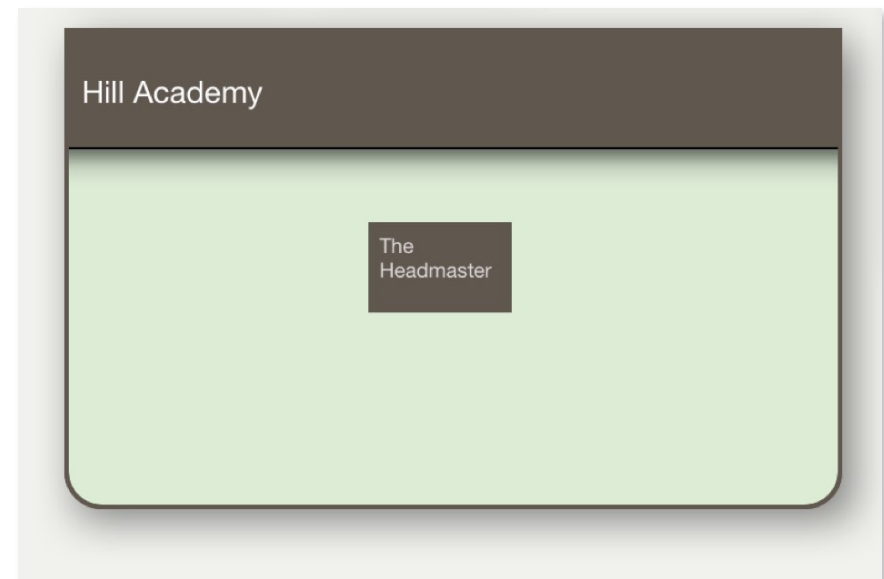
1. Select the tab that holds our map view of School Stories.
2. Create a new tab by pressing the “+” button at the right edge of the tab bar.

3. Press the left-most tab of the breadcrumb bar to move up to the top level of the document, or press the up-arrow key.
4. Double-click in the background to create a new note, “Hill Academy,” to hold our map. Resize this new note to make it quite large.
5. Every school story needs a head of school. Double-click the background of the map to make a new note named “The Headmaster,” and drag the headmaster inside “Hill Academy.”



## Changing the background

When a project contains several different map views, it can be helpful to change the background color of each map. Even a slight change can help remind you what part of the Tinderbox document you're currently working on.





1. Select the container named “Hill Academy.”
2. Open the Appearance Inspector, and switch to the “Bkgnd” pane.
3. Click the color well for the background color. The color picker will appear; replace the default light gray with a very light shade of green.
4. Close the color picker and the inspector.

## Inventing Geography

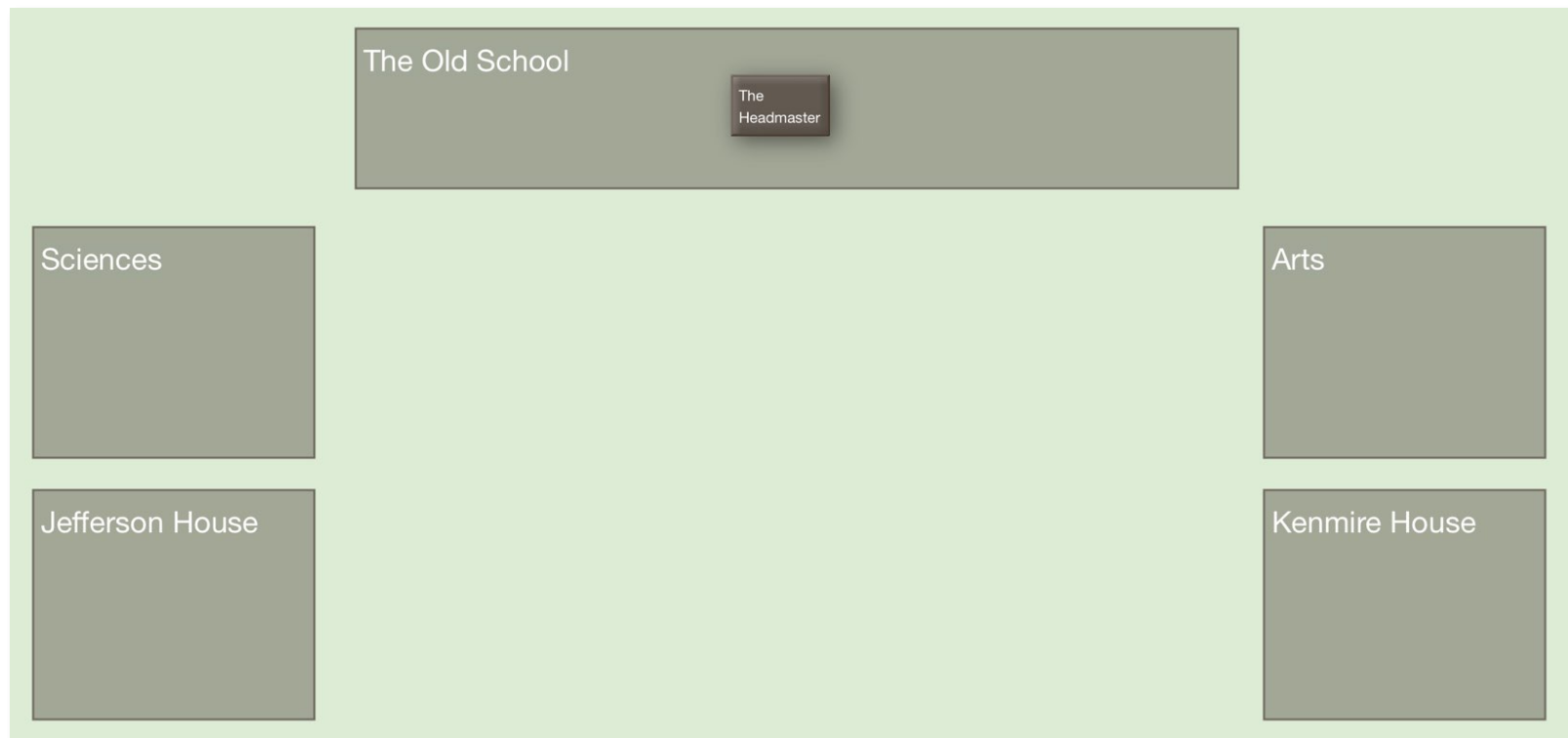
Next, let’s invent a world our characters will inhabit.

1. Select the container named “Hill Academy.”
2. Press the down-arrow key to zoom into the map view of the Academy.
3. If necessary, scroll the map window or drag its background to place The Headmaster near the center of the view.
4. Near the headmaster, right-click the background and choose **Create Adornment** from the contextual menu.
5. Name this adornment The Old School – the school’s central building. Resize the adornment so it will be quite wide; it’s a large and important building.
6. Drag “The Headmaster” to the center of The Old School.

On either side of the main building, we'll set a pair of smaller buildings that hold additional classrooms and instructional facilities. These, too, are adornments. We'll call one Arts and the other Sciences.

- ☞ You might find it convenient to change the scale of the map by choosing **Magnify** or **Shrink** from the View menu, or by using the pinch-to-zoom gesture on your trackpad.
- ☞ Pressing and holding the control, option, and command keys simultaneously will temporarily zoom out, providing an overview of the map.

Our school requires student dormitories. We'll add two: “Jefferson House” next to Sciences, and “Kenmire House” next to Arts.



## Reference Images

Our story of “Hill Academy” will trade on the mythos of the English Public School, as so many school stories do, though it will be set in a very different place and time. Still, a few reference images might be handy. We find, for instance, a Victorian photograph of Rugby at George P. Landow’s *The Victorian Web*.

1. In Safari, view <http://www.victorianweb.org/history/education/rugby/6.html>
2. Drag the photograph to the desktop, creating an image file.
3. Drag the image file into our Tinderbox map to make an *image adornment*. Adjust its position and size for convenience.
4. Add a note to contain the photo credit and any other information we might expect to need for this image.

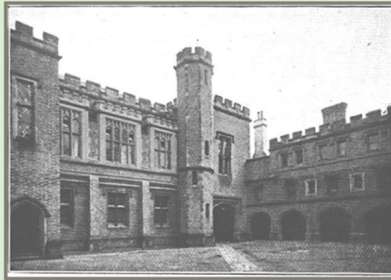


credit



credit





Rugby

credit



Swarthmore

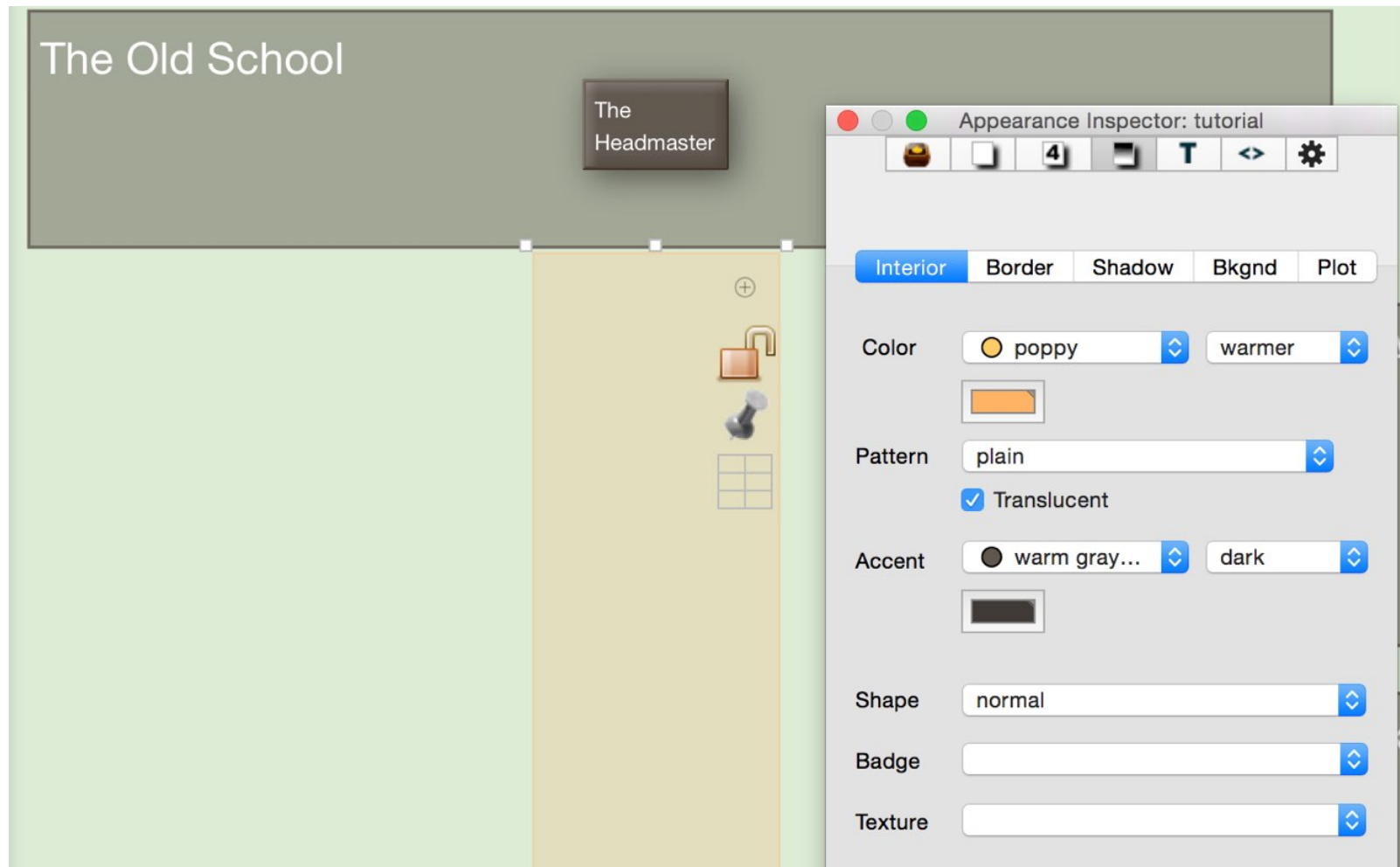
credit

A second reference image can be taken from an early view of the American Swarthmore College, <http://explorepahistory.com/displayimage.php?imgId=1-2-19CB>. We drag it into the map as well, resize appropriately, and add a caption and credit note.

## Captions

We might add captions to adornments (as well as other notes) to help annotate them.

1. Select the adornment for Eton.
2. Open the Text Inspector, and switch to the Caption pane.
3. Enter the caption “Eton” and press [return].



## Transparent Adornments

Transparent adornment can be useful for adding additional labels and annotations to maps where conventional captions are not sufficient.

1. Add a tall, narrow adornment to represent the path that runs from the central entrance of the Old School down the hill toward the nearby village. Give it no name.

2. Open the Appearance Inspector, and set the Color of the adornment to “poppy, warmer”.
3. Check the Translucent checkbox in reduce the opacity of the adornment.

We’ll now add a label for the “Great Lawn” – the grassy hillside that lies between the Old School and the village.

1. Right-click the background near the center of the lawn, and create a new adornment. Name is “Great Lawn”. Resize the adornment so it fills much of the lawn.
2. Open the Appearance Inspector, and set the Color of the new adornment to be “transparent.” Close the Inspector.
3. Switch to the Title pane Text Inspector.

4. From the **Map** menu of the Title pane, select 48pt text and centered alignment. From the Color pane of the same inspector, select “3” – a light gray.



5. Adjust the size and position of the adornment for convenience.

## Add Some Characters

Next, we'll add some characters to our story, creating a note for each new character. We'll begin with some instructors:

Mr. Henderson (Librarian)

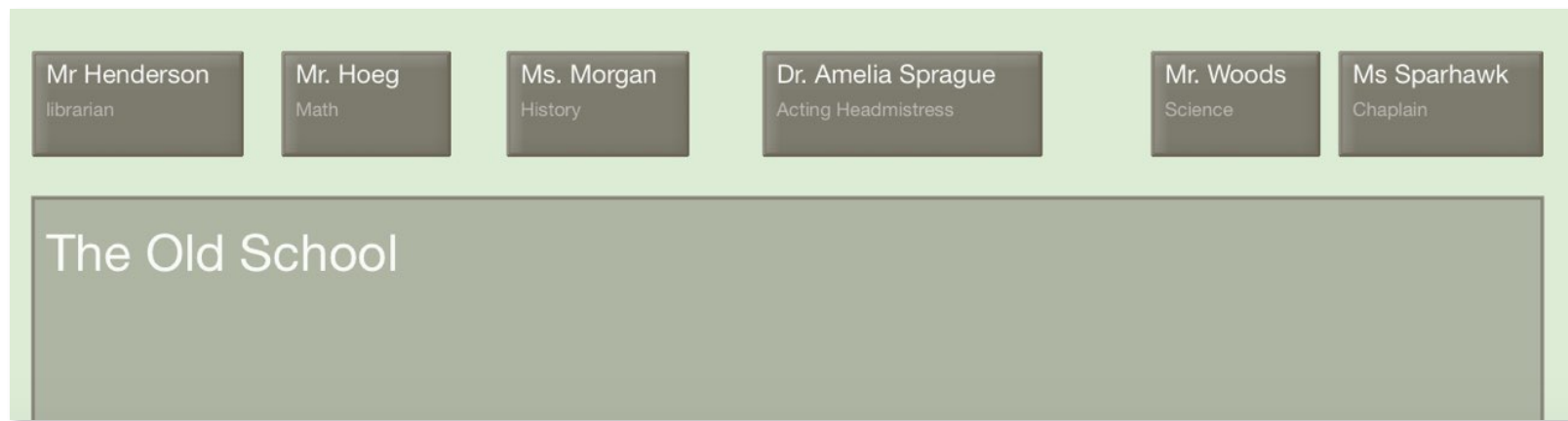
Mr. Woods (Science)

Mr. Hoeg (Math)

Ms. Morgan (History)

Ms. Sparhawk (Chaplain)

We'll place the character notes for instructional staff above The Old School, where they have their offices.



To remind us of their roles, we can add a subtitle to each note.

1. Select “Mr. Henderson.”
2. Open the Text Inspector, and switch to the Subtitle pane.
3. Enter the subtitle “Librarian” and press [return].

Repeat this for each instructor. For consistency, we also give our headmistress a name – Dr. Amelia Sprague.



Similarly, next to Kenmire House we add notes to represent that house's most notable characters: Amy, Cassie, May, and Jayden. Near Jefferson House, we add Trish, Mason, Isabella, and Noah. At the bottom the the lawn, where the Academy gives way to the Town, we add some incidental characters we expect to need:

Lucy Baker (tea shop)

Sue Gresham (newspaper reporter)

Officer Clevenger (security)

Turk (street urchin)

Aaron Naver (state prosecutor)

Sgt. Ozawa (resistance fighter)

## A Character Prototype

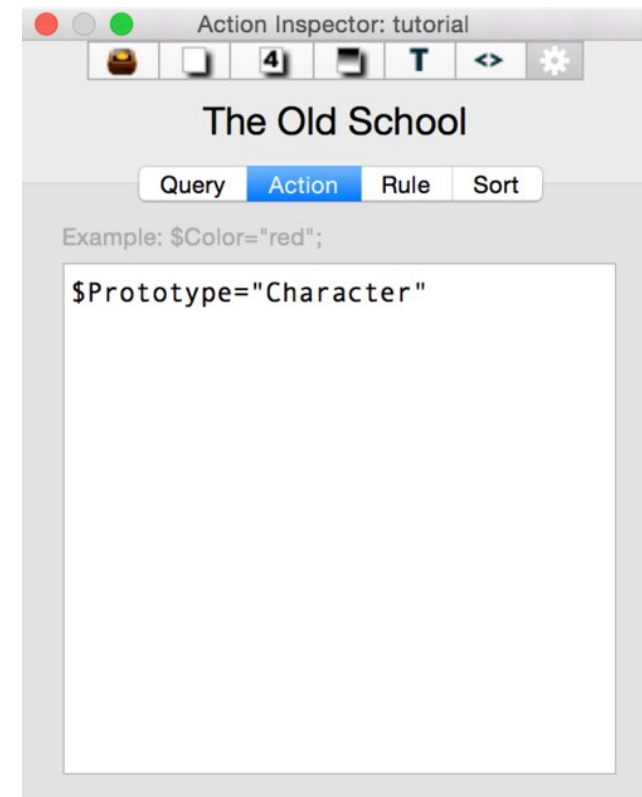
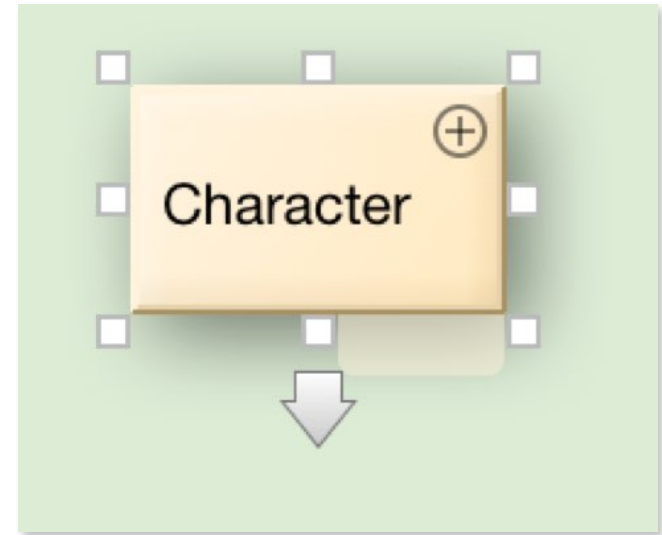
In just a few minutes of brainstorming, we've accounted for twenty characters. It might be useful to give character notes a distinct appearance; to do this, we'll create a prototype named Character they can all share.

1. Create a new note, Character, at the upper right of the map.

2. Open the Appearance Inspector, and select the Interior pane.
3. Let the Character's Color be: poppy muted.
4. Let the Character's Pattern be: diagonal gradient
5. Let the Character's Accent Color be: poppy light.
6. Switch to the Properties Inspector, and in the Prototype tab check "Prototype" to inform Tinderbox that Character can serve as a prototype.

At this point, we might select each character note in turn and use its prototype tab to set the note's Prototype to Character. This is not onerous, but we can make our task a little easier by adding an *adornment action*.

1. Select the adornment titled "The Old School".
2. Open the Action Inspector, and select the Action pane.
3. Add the action **\$Prototype="Character"** and press [return].



Now, whenever a note is dragged onto the adornment “The Old School”, that note will adopt “Character” as its prototype. The adornment action is performed when the note is moved onto the adornment; moving it away has no further effect. So, move each note in turn to the adornment, make it a Character, and then return it to its place.

For the students and townspeople, you might also use the Old School adornment, or you might add actions to Kenmire and Jefferson House as well.

## Stamps

If we expect to add lots of character notes, we can create a *stamp* to streamline the process. A stamp performs a Tinderbox action on the selected notes; all your stamps are listed in the **Stamp** menu so they’re always handy.

To create a new stamp:

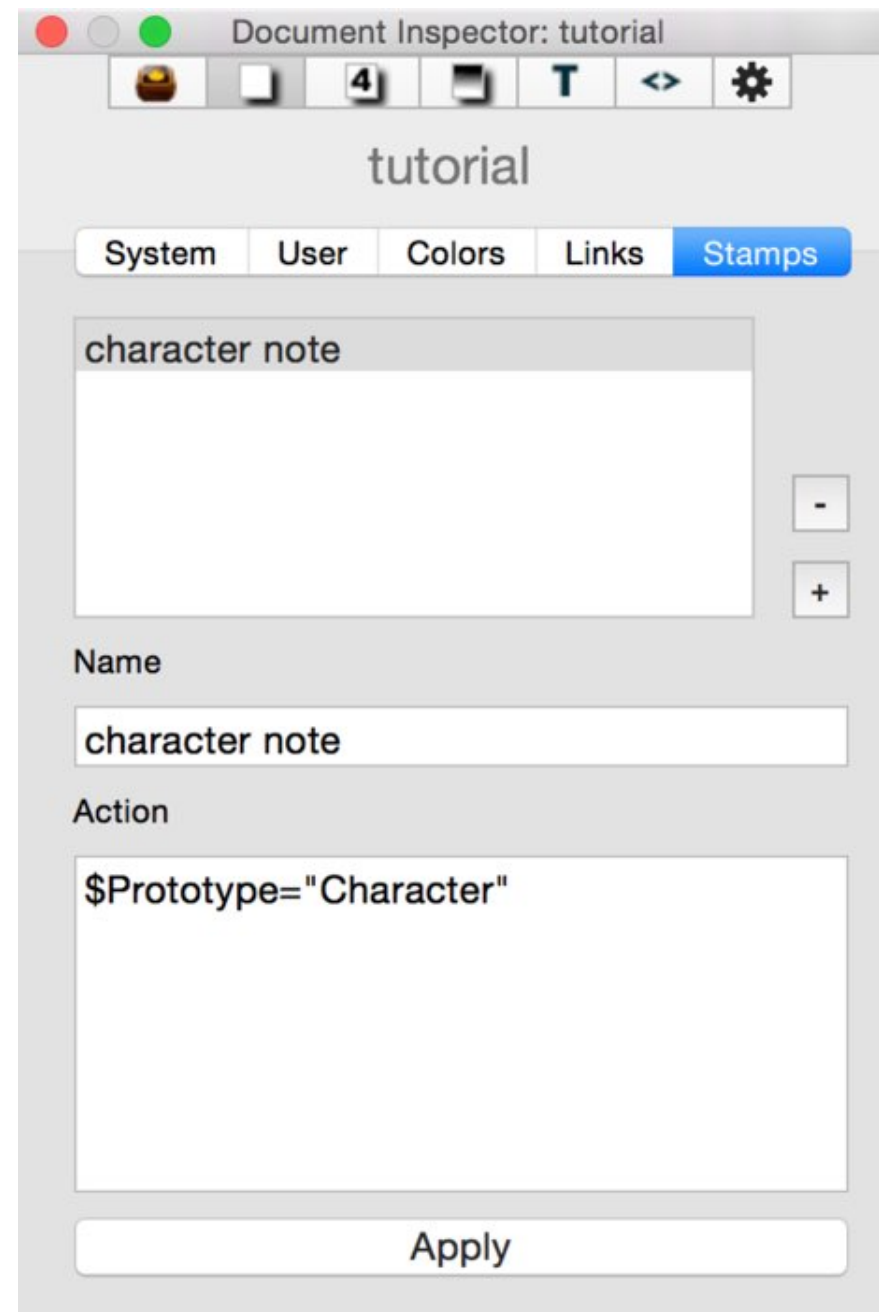
1. Select **Inspect Stamps...** from the Stamps menu to open the Document Inspector and select the Stamps pane.
2. Press the + button to make a new stamp.
3. Replace the placeholder “**stamp name**” with the name of the stamp: “character note”

4. Enter the action  
`$Prototype="Character"` and press  
 [return].

Your new stamp now appears in the Stamps menu. You can now select any note or group of notes and then apply this stamp to make them all Characters.

- ☞ A convenient way to select several notes in map view is to press and hold down the option key. The cursor changes to a crosshair; drag to select any rectangular region.
- ☞ You may also add notes to the current selection by pressing the ⌘ key while selecting additional notes.

Note that stamps are only applied when you apply them. OnAdd actions are applied automatically when you add a note to a container or move it onto an adornment. Because



they're automatic, OnAdd actions can be easy to forget about; they just happen, organizing things automatically.

## 14. Brainstorming With Maps and Grids

Our project might require a research trip to England to visit some historic schools, to see their environs, and perhaps also to meet with colleagues who might give fresh insights into the project. Travel is costly and time-consuming, so extra planning may prove rewarding. Just as important, plans change and opportunities arise unexpectedly; preserving the malleability of your planning helps adjust to the unexpected.

### Creating a Planning Container

As before, we'll make a new tab for our travel plans, and create a new container to hold them.

1. Select the tab that holds the Map view of “Hill Academy.”
2. Press the “+” button at the right edge of the tab bar to create a new tab.
3. Press “tutorial” at the left edge of the Breadcrumb bar to move to the top level of the document.
4. Create a new note, “Travel Plans,” and enlarge it to be roughly the same size as “Hill Academy”

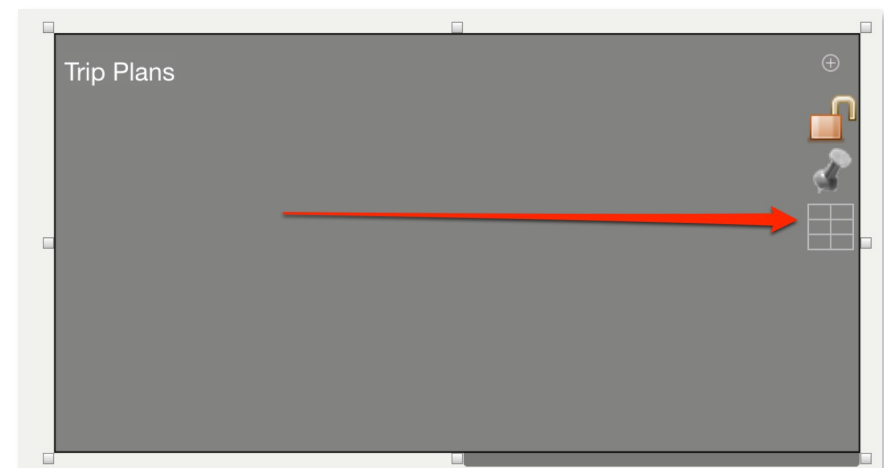
5. Select the new “Travel Plans” and press down-arrow to zoom into the (empty) container.

## Creating a Time Grid

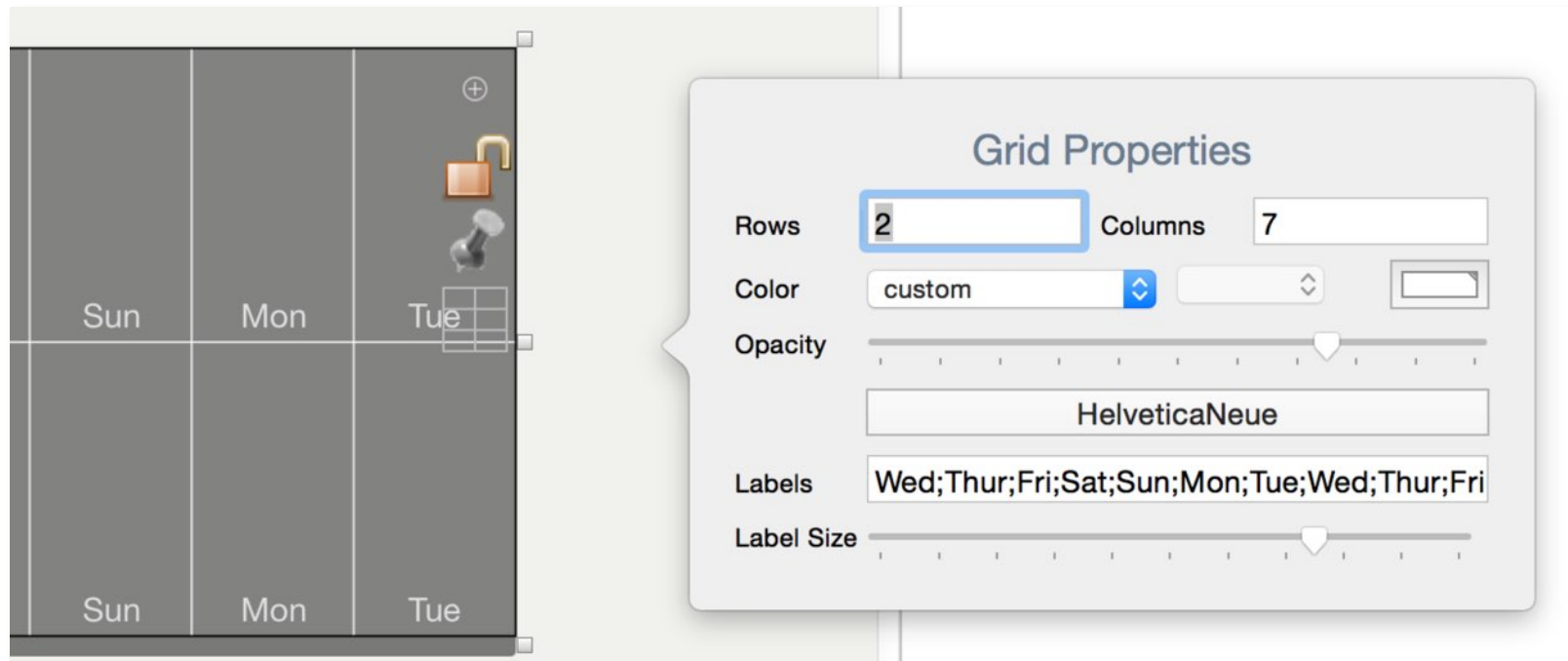
We don't yet know precisely when we will take this trip, which will probably be most useful several months from now, after we are farther along in the project. Still, we can establish rough parameters. Certainly, the long plane trip suggests that we'll want to spend at least a week, and other commitments make a trip longer than a month difficult to imagine. For planning, we'll suppose we have two weeks at our disposal.

We'll begin by adding an adornment to serve as an informal calendar grid.

1. Right-click in the background of the map and choose Create Adornment from the contextual menu.
2. Name the new adornment, “Trip Plans”
3. Using the Appearance Inspector, set the color of the new adornment to "black"
4. Select “Trip Plans” and press the Grid button.



5. A visit to some airline web sites suggest that fares will be least costly if we leave on Tuesday or Thursday. In the Grid popover, request two rows, seven columns, and labels “Wed;Thu;Fri;Sat;Sun;Mon;Tue;Wed;Thu;Fri;Sat;Sun;Mon;Tue;”.
6. Adjust the Label Size for legibility.
7. Click outside the popover to dismiss it, and resize the adornment so it is conveniently large, filling much of the display.
8. For convenience, click the lock button of the adornment to lock its size and position. If we want to move it, we can always unlock it later.





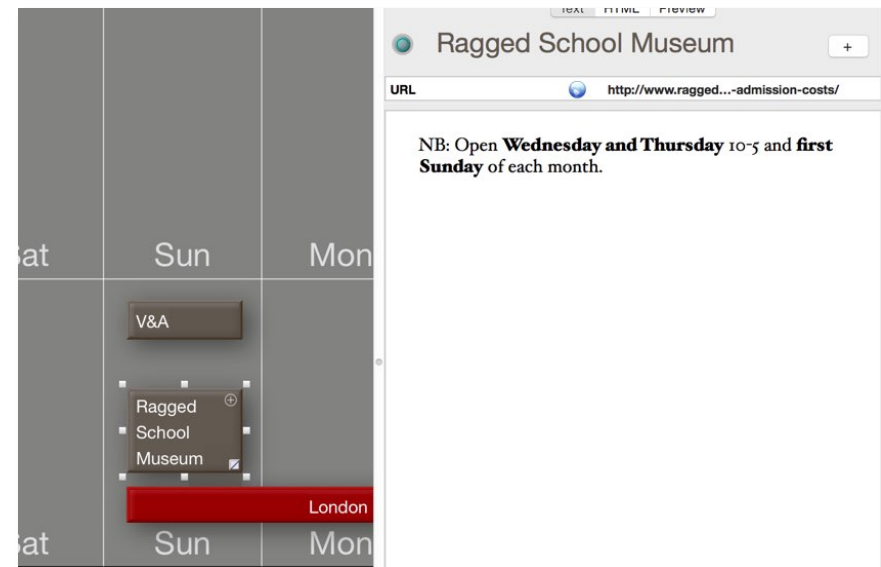
# Discovering Constraints

Let's proceed to sketch out how we might spend time during this trip. If we depart on Tuesday, we'll arrive in London on Wednesday morning. We'll need to return to London by the following Monday, since we expect to depart sometime Tuesday. We can easily add notes to remind us of these constraints.

The image shows a travel planning interface. On the left is a 'Trip Plans' grid. The grid has columns for days of the week (Wed, Thur, Fri, Sat) and rows for different types of information (agent, attributes, book, info, map, similar, text, uri). A red button labeled 'London' is in the top-left cell. On the right is a map window titled 'Schools Museum, Hitchin, UK' showing a map of the region around London and Hitchin, with various towns and roads labeled.

We will want to spend several days in London and its environs.

Observing that we need to be in London on Monday, the preceding Sunday might be a good opportunity to visit museums. Resize the second London note to span Sunday and Monday.



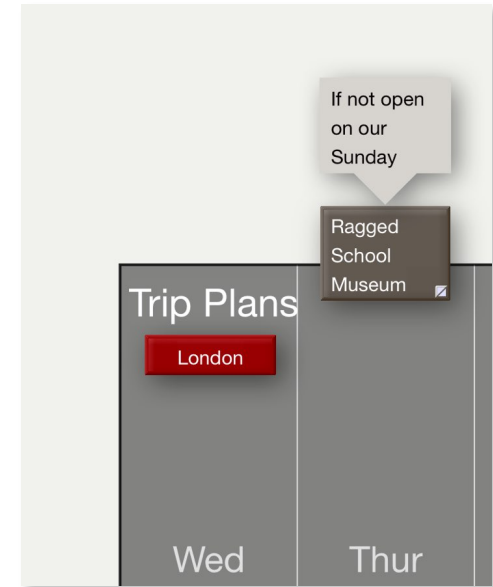
On Sunday, we'll pencil in trips to the Victoria and Albert Museum and the Ragged School Museum. For the latter, we make a note of the museum's URL in the \$URL attribute: <http://www.raggedschoolmuseum.org.uk/>. We also add a text reminder that the museum is open Wednesdays or Thursdays and on certain Sundays only. This either constrains our date of departure or means that we'll need to visit this museum at the beginning, rather than the end, of our trip.

Continuing to work backwards in our itinerary, a second school museum is located in Hitchin, not far from London and also convenient for a tour of Harrow. To confirm that this makes sense, we add a note for the Schools Museum, open the Get Info window, and enter its approximate address,

which will be stored in the note's \$Address attribute: "Schools Museum, Hitchin, UK". Tinderbox locates the town; we'll pencil this in for Saturday.

## Adding Memoranda

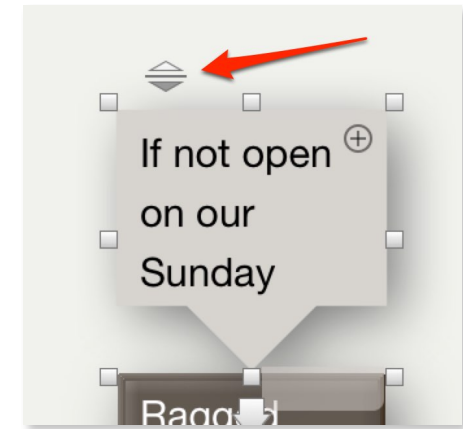
If often proves useful to add notes that remind us of uncertainty or contingency. Though our trip to the Ragged School Museum is pencilled in for the second Sunday of our trip, it might be necessary to move it to a Wednesday or Thursday. We can add a second note or an alias of the first note about the Ragged School Museum to represent this contingent commitment.



The “tag” shape in our memorandum is often useful. To create this note:

1. Create a new note named “If not open on our Sunday” and position it above the note on which it comments.
2. Using the Interior pane of the Appearance Inspector, change its color to “lightest warm gray dark”.
3. In the same pane, change the note’s shape from “normal” to “vertical tab.”

4. The new note's tab may point upward rather than downward. To switch orientations, click on the *orientation* widget above the note.



## Informal Lists

We might proceed by jotting down a number of things we might perhaps want to visit in the course of our expedition. There will be people we should try to visit – experts in the field, colleagues, old friends. We might want to visit a number of schools, museums, and galleries. We might also want to scout locations that could be useful in the course of our book, as well as booksellers and libraries which may have work we should consult.

Later, as our plans become clearer, we can move some of these to the time grid. Others might be

people	schools	museums	places
Jonathan	Rugby	British Schools	Haye on Wye
Elisa	Harrow	Ragged Schools	a moor dark, stormy
Elin	Eton	V&A	London Library
Soton faculty	Winchester	Ashmolean	
Keith	Oxford	Soane Museum	
Mark A	Cambridge		

clustered together; if we visit Oxford, we'll certainly visit the Ashmolean. Mark A lives in Portsmouth, which is not far from Southampton; the Soane Museum is not very far from the London Library. Often, we'll want to move gradually from simple lists to informal clusters, and then move those clusters into our schedule. Occasionally, we might swap one cluster for another, forgoing a day in Bristol in order to arrive earlier in Haye on Wye.

## 15. Explode

As we continue to research and plan, a helpful friend sends encouraging email.

Yes – do visit Haye-on-Wye. And while you're in the West Country, there's lots of good food!  
Off the top of my head:

Nathan Outlew's places in Cornwall (at Rock and other places <http://www.nathan-outlaw.com/#dialog>)

Rick Stein in Padstow, Cornwall – fish again

Carved Angel in Dartmouth, Devon

Jamie Oliver has a place in St Ives that was well written up

Hix has a fish place at Lyme Regis, Dorset

We copy the list of restaurants into a note in our travel plans. But we don't really want a note about five restaurants; we want five notes, each describing one restaurant which we might be able to visit.

Since there are only five restaurants in this list, we could create five notes without much difficulty.

👉 In fact, we could select each paragraph of text in the text pane and drag it into the map pane to create a note with that text as its title.

But if we had twenty notes to create, this might be tedious, and if we had 100 notes or more, we'd be tempted to summon the intern.

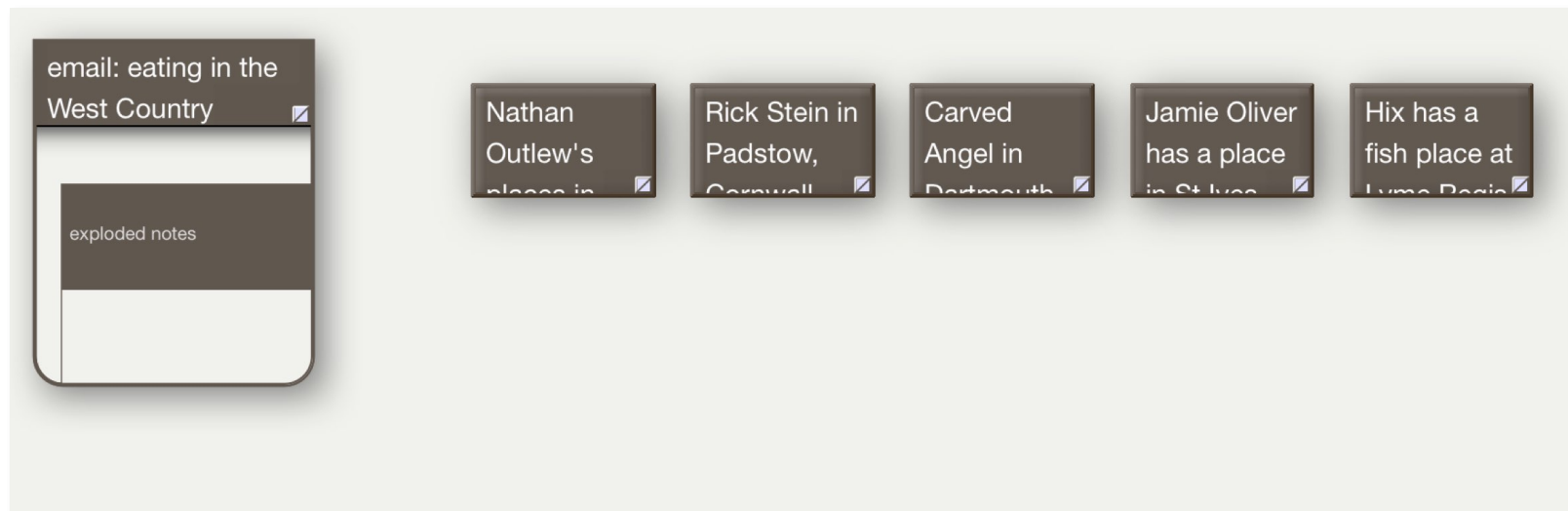
We can automatically explode the note into separate notes by selecting the note of interest and choosing **Explode...** from the Note menu.

The screenshot shows a dialog box titled 'Explode' with the following elements:

- Two radio buttons: ☒ Break at paragraphs and ☐ Break at delimiter.
- A 'Title' section with a dropdown menu showing 'First Sentence' and two checkboxes: ☐ Remove title from text and ☐ Omit text.
- An 'Example' section showing a preview of the resulting notes. The first note has the title 'Nathan Outlew's places in Cornwall (at Rock and other...' and the body 'Nathan Outlew's places in Cornwall (at Rock and other...'.
- A large 'Explode' button at the bottom left.
- A pagination control at the bottom right showing '1 of 5 notes' with up and down arrows.

We now have a container – our original note – which holds a container of five exploded notes: one for each suggested restaurant. Zoom into the container, and then zoom into the container of exploded notes; you’ll find a note for each paragraph in the original email.

Next, we’ll copy the exploded notes and paste them into our map. (Typically, we’d simply move them; in this case we make duplicates so you can see the original results.)



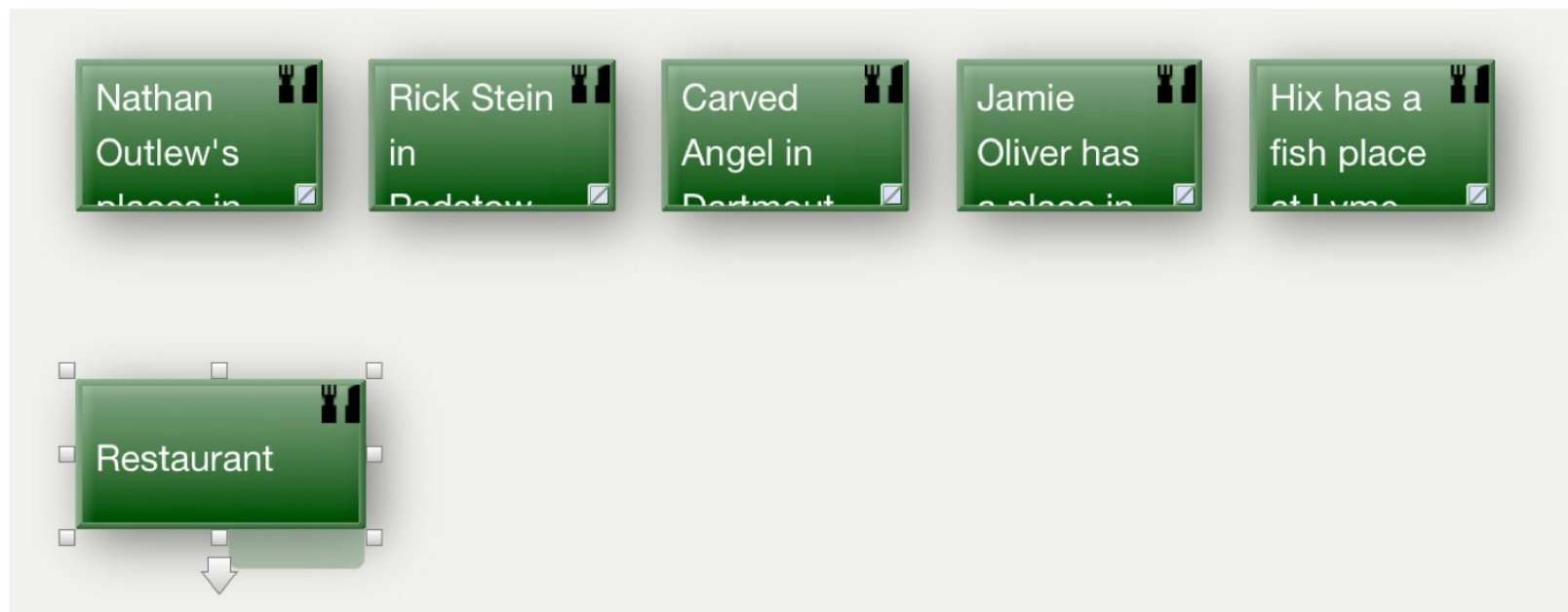
Next, we’ll create a new prototype note, “restaurant”, for places we might like to eat.



A “restaurant” is pretty much like any note. We want to use it as a prototype, so we select “restaurant” and check “Prototype” in the Prototype pane of the Properties Inspector.

Turning to the text pane of “restaurant”, we add some key attributes by pressing the “+” button in the upper right-hand corner of the text pane. Many restaurants will have a URL, so we’ll add \$URL as a key attribute. We’ll add some additional key attributes: \$Address, \$Email and \$Telephone. And we’ll give restaurants a distinctive color as well.

Finally, we select the new notes and, clicking on one of their prototype tabs, choose “restaurant”. All five restaurants receive their new prototype. Having



done this, it's easy work to visit each note in turn and to fill in the restaurant's URL, address, and contact information.

In this way, we've taken an informal list of restaurants and quickly converted it into separate notes. Each note has some metadata, and each shares a common prototype. Because all these restaurants share a prototype, it will be easy for agents (and for you) to find all the restaurant notes or to add additional information that all restaurants should share.

For example, we might well want to add another key attribute to remind us who recommended the restaurant. We'll might create a new attribute, \$Source, in the Document inspector. If we add Source to the key attributes of the prototype "restaurant," Source will become a key attribute for each of our newly-created restaurants.

