RIA Patterns

Best Practices for Common Patterns of Rich Interaction

Bill Scott
Yahoo! Ajax Evangelist
b.scott@yahoo.com
Origin of Patterns

A Pattern Language, Alexander, 1977

Dissatisfied with sterile, impersonal design

Defined it as
Solution to a problem in context

Captured
Situation, competing constraints, canonical solution

Examples
Couples Realm, House for a Couple, Sitting Circle, Marriage Bed
Origin of Patterns

Applied to software design, Gang of four, 1995

1996, Alexander raises the challenge higher

User experience patterns

1997, Tidwell’s Common Ground

Welie’s pattern site

Design of Sites book

2005, Tidwell, Designing Interfaces
Welcome
Welcome to the Yahoo! Design Pattern Library. We hope to be a monthly resource that is full of patterns for
software development.

What's a Pattern?
A pattern describes an optimal solution to a problem. It includes an architecture, an implementation,
and a description of why and when to use it.

Recent Patterns
NATIVITY
- Navigation
- Fly-out Menus
- Hub and Spoke
- Left Navigation
- Tabs
- Module Tabs
- Navigation Tab

EXPLORATION
- Auto Complete
- Calendar Picker
- Pagination
- Item Pagination
- Search Pagination

ORGANIZATION
- Data
- Feedback
- Cart
- Shipping
- Support
- Register
- Login

CUSTOMIZATION
- Drag and Drop
- Drag and Drop
- Reuse
- In Place Editing
- Filters

APPLICATION
- Call to Action
- Improve Readability
- API
- Review Architecture
- Group Related Items
- Organize Screen/Page

Results Page Next
Search Pagination
Rating an Object
The user needs to select a set of search results ranked by
relevance that is too large to easily display within a single page.

Restaurant
A user wants to quickly leave their opinion on an object, with minimal interruption to any other
task they are involved in.

Solution
- Show clickable items (most often used are stars) that light up on rollover to infer clickability.
ria patterns

invitation. cursor invitation. hover invitation. tooltip invitation. page invitation. drop invitation. available. selected. auto complete. live suggest. refining search. dynamic filtering. live search. on-demand scrolling. deferred content loading. fresh content. on-demand content. periodic refresh. resizable modules. scrolling modules. hover detail. narrowing actions. narrowing choices. auto form fill. in-place drill down. dependent choices. look before you leap. drag and drop. drag and drop modules. drag and drop objects. persistant portals. in page edit. inline text edit. inline custom edit. direct state edit. rating an object. inline custom edit. inline tag edit. popup custom edit. grid cell edit. slide out custom edit. module configure. module flip configure. module inline configure. module slide out configure. module faceplate. expandable content. in-context tools. in-context links. in-context hover menu. silent submit. remembered collection. auto save. remembered preferences. hover spy. inline assistant. inline validation. validate then suggest. indicator. busy indicator. progress indicator. inline status. in-context busy. cursor busy. opacity fade. high contrast. balloon error tip. dynamic goal. opacity focus. detail zoom. lightbox. transition. brighten. cross-fade. dim. expand. fade-in. fade-out. flip. move. self-heal. collapse. slide. animate.
Anatomy of an RIA Pattern

• Interaction
  event, timer, invitation

• Operation
  lookup, action, validate, message

• Presentation
  update to the interface
interaction.

- Every pattern starts with an interaction
  User event (event)
  System event (timer)

- Some patterns are about **inviting** interaction

  mouseout  hover  keypress  keydown  mousedown
  drop  filter choices  mouseup  drag  click  mousedown  select
  focus  blur  resize  move  timeout  select

  Interaction  Operation  Presentation
operation.

- Lookup. I can get information when I need it.
- Action. I can take action in context.
- Validate. I can prevent errors early.
- Message. I can communicate instantly.
operation. lookup.

- I can get information when I need it

operation. action.

• I can take action in context

operation. validate.

- I can prevent errors early

inline validate. validate then suggest. balloon error tip. character counter. live preview.
Anatomy of an RIA Pattern

- Interaction

Anatomy of an RIA Pattern

- Interaction

Interaction + Operation + Presentation = Ajax Pattern

Interaction
Operation
Presentation
patterns. invitation.

• Problem
  • The designer needs to cue the user that an interactive feature exists.

• Use When
  • You want to invite the user to click or interact with the object being hovered over.
  • You want to make it clear that something will happen when the user clicks on the object being hovered over.
  • You want to entice the user to interact with the feature.
invitation. cursor invitation. hover invitation. tooltip invitation. page invitation. drop invitation.
Potential Pitfalls
  • discoverability

Best Practices
  • use hover to reveal interaction
  • use the familiar to teach the new
  • use tours sparingly
  • make invitation engaging, but think of how it will interact with rest of page
pattern. hover + cursor + page + tooltip + drop invitation

Potential Pitfalls
- discoverability

Best Practices
- use hover to reveal interaction
- use the familiar to teach the new
- use tours sparingly
- make invitation engaging, but think of how it will interact with rest of page
- keep actions out of hover
- don’t proselytize
- reveal invitation instantly
- don’t create page jitter

To see a product preview in action, hover over the link below.

Forty Associates recently added product previews to their Web site as part of a small beta program. During the beta, visitors to Associate sites who viewed a product preview clicked through to Amazon.com over 4% of the time, those clicks resulted in a purchase nearly 6% of the time.

Now we’re opening the beta up to all Associates to further improve product previews. The first 500 Associates who refer a sale through a product preview-enhanced link will receive a $5 Amazon gift certificate. Click here to join your fellow Associates in beta testing product previews.

Don’t worry if you don’t already have Basic Display Product Links on your Web site. You can create them as part of the sign-up process. Click here for answers to questions you might have about the beta program.
patterns. selection.

- **Problem**
  - The user needs to choose objects to operate on.

- **Use When**
  - You to provide a way to mark an object or objects as actionable.
patterns. selection.

item selection. object selection.
pattern. item selection.

**Use When**
- within a paging context
- need a simple way to provide discontiguous selection

**Potential Pitfalls**
- confusion between checkbox and clicking in row
- mixing with drag and drop
- handling actions on no selection

**Best Practices**
- use only within context of paging; not for scrolled content
- combine with a row of buttons or toolbar that operates on the selected items
- use light shading to re-enforce selected state
- avoid using with drag and drop
pattern. item selection.

**Use When**
- inside a paging context
- need a simple way to provide discontiguous selection

**Potential Pitfalls**
- confusion between checkbox and clicking in row
- mixing with drag and drop
- handling actions on no selection

**Best Practices**
- use only within context of paging; not for scrolled content
- combine with a row of buttons or toolbar that operates on the selected items
- use light shading to re-enforce selected state
- avoid using with drag and drop at the least avoid allowing drag of non-selected objects
- display prominently the number of items selected
pattern. object selection.

**Use When**
- in a scrolled table
- selecting objects

**Potential Pitfalls**
- complexity of contiguous vs. discontiguous
- using CTRL for discontiguous will not work on Macintosh

**Best Practices**
- allow multiple selection
- allow discontiguous selection
pattern. auto complete.

• **Problem**
  
  • The user needs to enter an item into a text box which could be ambiguous or hard to remember and therefore has the potential to be mis-typed.

• **Use When**
  
  • The suggestions can be pulled from a manageable set of data.
  
  • The input item can be entered in multiple ways.
  
  • The input item can be matched with a specific data item in the system.
  
  • Speed and accuracy of entry is an important goal.
  
  • The total number of items would be too large or inconvenient for displaying in a standard drop down box.
Potential Pitfalls
- displaying completions too soon
- displaying completions too often

Best Practices
- don’t require the user to arrow or mouse to the selected item
- tab should select the best match
- delay popup until user slows typing enters a known delimiter
- if possible show character match with bold type
- if too many matches, then show truncated list of best matches
patterns. lookup content.

live search. refining search. on-demand scrolling. deferred content loading. periodic refresh. on-demand refresh. hover detail.
Use When
• user needs to search for content and are uncertain on the correct keywords.

Potential Pitfalls
• if results are returned too quick, will be distracting
• if results are not returned quick enough, it will feel sluggish

Best Practices
• start returning results when the user “slows down” typing
• show results below text entry field for feedback
pattern. live search.

**Use When**

- user needs to search for content and are uncertain on the correct keywords.

**Potential Pitfalls**

- if results are returned too quick, will be distracting
- if results are not returned quick enough, it will feel sluggish

**Best Practices**

- start returning results when the user “slows down” typing
- show results below text entry field for feedback
pattern. refining search.

Use When
- user needs to refine a search
- for merchandise search

Potential Pitfalls
- sluggish performance

Best Practices
- place refining criteria to left of results
- use checkboxes for toggling filters
- use sliders for value ranges
- generally avoid sliders for single values (can combine slider & input)
- provide a “show all” to undo refinement
- try to keep criteria above the fold
pattern. on-demand scrolling.

**Use When**
- chunking data would affect user flow
- user feels “ownership” of data
- content is data; not search results
- data content will be sorted, filtered, etc.
- selection model is continuous

**Potential Pitfalls**
- dual-scrollbar issue
- sluggish performance
- extremely large data sets

**Best Practices**
- provide dynamic tooltip showing location within scroll
- animate scroll
- if desire a hybrid, use animation on paging.
Use When
- some peripheral content might be slow to load

Potential Pitfalls
- loading indicator can be distracting

Best Practices
- only apply sparingly to improve performance on main pages
- use for external content
pattern. periodic refresh.

**Use When**
- content is based on live information
- for enticing users to click-through

**Potential Pitfalls**
- can be distracting if not primary

**Best Practices**
- if secondary to page, then make refreshes less frequent and simple
- if primary to page, then make refreshes more frequent & visible
pattern. hover detail.

Use When
- need to show detail in context

Potential Pitfalls
- popup too quick
- popup too slow, seem sluggish

Best Practices
- show detail after 0.25 second delay
- dismiss immediately on mouse out or click
patterns. dragdrop.

• **Problem**
  - The user needs a way to perform operations on one or more objects by being able to move objects from one place to another directly with the mouse.

• **Use When**
  - You want to avoid forcing the user to go to another page in order to re-arrange objects.
  - For re-arranging content modules, re-ordering lists, changing containment.
patterns. dragdrop.

drag and drop.
drag and drop modules.
drag and drop objects.
pattern. drag and drop.

**Use When**
- you want to manipulate a natural visual model of objects

**Potential Pitfalls**
- starting with drag & drop
- using it to set a single attribute
- constructing artificial visual constructs
pattern. drag and drop.

**Use When**
- you want to manipulate a natural visual model of objects

**Potential Pitfalls**
- starting with drag & drop
- using it to set a single attribute
- constructing artificial visual constructs
- using it for removal
- confusing drag ghost with drag object
- user's don't get it

Drag photos here to edit them as a batch. You can then change any photo attributes or create a new set.

---

**Pick Up at the Store**

- Chips
- Milk
- Bread
- Orange Juice2
- Tomatoes
- Lettuce

---

- What is spam?
- Can I transfer my message credits?
- How do I delete my HandyPay account?
- What if I didn't receive my full message package?
- I need to contact customer care regarding the Compose Text Message service.
pattern. drag and drop.

**Use When**

- you want to manipulate a natural visual model of objects

**Potential Pitfalls**

- starting with drag & drop
- using it to set a single attribute
- constructing artificial visual constructs
- using it for removal
- confusing drag ghost with drag object
- user's don't get it
- creating page jitter
pattern. drag and drop.

**Use When**
- you want to manipulate a natural visual model of objects

**Potential Pitfalls**
- starting with drag & drop
- using it to set a single attribute
- constructing artificial visual constructs
- using it for removal
- confusing drag ghost with drag object
- user's don't get it
- creating page jitter

**Best Practices**
- use for changing layout
  - minimize movement on drag operation
  - balance preview on drag with page jitter
- use for re-arranging lists
  - if using ghost, use insertion bar
  - if direct drag, don't use ghost
- start drag after 3 pixel delta
- change cursor over drag candidate
- use drop invitations
- signal valid/invalid targets
- use zoom back
- allow ESC to cancel; outer bounds
- take care of interesting moments
- use transitions to smooth out effect
Drag and Drop Modules - Interesting Moments Grid

Currently on beta.my.yahoo.com

ID: Bill Scott & Eric Miraglia

Date: Mouse Hover

Mouse Down

Drag Initiated

Drag Over
Valid Target

Drag Over
Invalid Target

Drop Parent

Cursor

CSS Move cursor

CSS Move cursor

CSS Move cursor

CSS Move cursor

Tool Tip

Drag Object

Message Center

Check Email
Launch Messenger 15 friends on

Full Opacity

Reduced Opacity

Reduced Opacity

Reduced Opacity & Invalid Badge

Reduced Opacity

Drop Target

Message Center

Check Email
Launch Messenger 15 friends on

No insertion bar, just a gap

No insertion bar, just a gap

Insertion bar showing where it will drop

No insertion bar, just a gap

No insertion bar, just a gap

Message Center

Check Email
Launch Messenger 15 friends on
<table>
<thead>
<tr>
<th>Page Content</th>
<th>Mouse Hover</th>
<th>Drag Initiated</th>
<th>Drag over Valid</th>
<th>Drag over Invalid</th>
<th>Drag over Original</th>
<th>Drop Accepted</th>
<th>Drop Rejected</th>
<th>Drop on Original</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hint</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Cursor</td>
<td>Normal</td>
<td>Move Cursor</td>
<td>Move Cursor</td>
<td>Move Cursor</td>
<td>Move Cursor</td>
<td>Normal</td>
<td>Normal</td>
<td>Normal</td>
</tr>
<tr>
<td>Drag Object</td>
<td>Normal</td>
<td>Normal</td>
<td>Reduced Opacity &amp; Tracking</td>
<td>Reduced Opacity &amp; Tracking</td>
<td>Reduced Opacity &amp; Tracking</td>
<td>Normal</td>
<td>Normal</td>
<td>Normal</td>
</tr>
<tr>
<td>Orig Location</td>
<td>Normal</td>
<td>Normal</td>
<td>Hole Opens</td>
<td>Hole Remains</td>
<td>Hole Remains</td>
<td>Hole Remains</td>
<td>Hole Remains</td>
<td>Hole Remains</td>
</tr>
<tr>
<td>Drop Target</td>
<td>Normal</td>
<td>Normal</td>
<td>Normal</td>
<td>Insertion Bar</td>
<td>N/A</td>
<td>1. Insertion Bar Removed</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Pattern: drag and drop.
patterns in page edit.

- **Problem**
  - User needs to edit content directly on the page.

- **Use When**
  - You want to avoid having the user move to a different page to perform an edit.
  - The edits are best performed in context of the page.
patterns. in page edit.

inline text edit. inline custom edit. popup custom edit. in form edit. direct state edit.
rating an object. inline tag edit.
grid cell edit.
pattern. inline text edit.

Use When
- directly editing text

Potential Pitfalls
- discoverability
- placing buttons elsewhere on page
- edit & render modes different sizes
- confusing highlight with selection
- too subtle distinction of edit mode
- communicating changes made

Best Practices
- avoid page jitter
- make render & edit modes same size
- activate on click
  - alternatively use in-context tools
- deactivate on blur
pattern. inline custom edit.

**Use When**
- item being edited is multi-field
- you want to be explicit about the edit mode

**Potential Pitfalls**
- discoverability
- too subtle of invitation
- clutter visual display w/ invitations
- making page jump
Use When

- item being edited is multi-field
- you want to be explicit about the edit mode

Potential Pitfalls

- discoverability
- too subtle of invitation
- clutter visual display w/ invitations
- making page jump
Pattern: inline custom edit.

**Use When**
- item being edited is multi-field
- you want to be explicit about the edit mode

**Potential Pitfalls**
- discoverability
- too subtle of invitation
- clutter visual display w/ invitations
- making page jump
Use When

- item being edited is multi-field
- you want to be explicit about the edit mode

Potential Pitfalls

- discoverability
- too subtle of invitation
- clutter visual display w/ invitations
- making page jump

Best Practices

- activate with incontext tools
- use traditional form (with save button)
- smooth out open/close transition
- use invitations to clue user they can edit
- provide feedback during save
pattern. popup custom edit.

**Use When**
- Context is not critical
- Information is ancillary

**Potential Pitfalls**
- Using a popup when inline would work best
- Hiding the activator on popup
- [more]

**Best Practices**
- Popup should appear next to activator
- Activate with incontext link
- Provide a form for editing
- Use standard submit button, like save
patterns. in form edit.

Use When
- editing a formatted area
- content will be affected by format

Potential Pitfalls
- [more]

Best Practices
- [more]
pattern. inline tag edit.

Use When

- more

Potential Pitfalls

- will create a closed vocabulary
- only as good as suggestions
- confusion on commas vs. space
- confusion on my choice vs. suggest
- confusion on singular/plural, etc.

Best Practices

- allow tags to be separated by commas
- allow suggested tags to be entered/removed with click
Use When

- you want to provide a way to directly change a rating
- you want to increase size of creator community

Potential Pitfalls

- confusion between community rating and personal rating
- after voting, what is the community rating?
- providing too many choices
- how user changes choice

Best Practices

- provide as few choices as possible
- use stars for 1 of n choices
- use thumbs up or single button for binary rating
pattern. direct state editing.

**Use When**
- flagging objects in a list or grid
- bookmarking favorite

**Best Practices**
- provide a hover invitation
- provide a unflagged state as well as a flagged state
patterns. in-context tools.

• Problem
  - User needs to perform actions on specific objects on the page

• Use When
  - It would be too busy to place actions visibly beside each object.
  - You don’t have a way to select objects and perform actions on selection.
  - You don’t need to perform the actions on more than one object at a time.
  - The actions are best performed in context of the page.
patterns. in context tools.

in-context links. in-context menu.
pattern. in-context tools.

**Potential Pitfalls**
- can be tedious if multiple objects needed to be acted on
- visually cluttered if actions are always visible
Potential Pitfalls

- can be tedious if multiple objects needed to be acted on
- visually cluttered if actions are always visible
- discoverability if reveal on hover

Pick Up at the Store
- Milk
- Orange Juice
- Tomatoes
- Lettuce
- Bread
- Chips
Potential Pitfalls
• can be tedious if multiple objects needed to be acted on
• visually cluttered if actions are always visible
• discoverability if reveal on hover

Best Practices
• use when it is common to operate on individual items
• provide ways to also operate on the whole group
• use familiar constructs (hyperlinks for actions) to teach new idiom
• use in-context busy indicator if action will take some time to complete or use yellow fade technique to signal action completion
• consider the visual space for the tools when designing the items
• use hover, cursor, tooltip invitations to signal actionable area
• don’t mix selection-based actions with in-context tools
patterns. persistence.

remembered collections. auto save.
Use When
- desire to minimize “bail-out” moments
- creating shopping experiences
- for bookmarking

Possible Pitfalls
- connection between remembered collection and items could be lost on user

Best Practices
- attach the remembered collection to the shopping cart, list
- reveal the contents in context to the page (slide, instant reveal, etc.)
- bring the collection in proximity to the shopping experience
- make sure the connection is clear between saving the object and it entering into the collection
pattern. auto save

Use When
- composing email messages
- composing web documents
  (spreadsheet, text)

Possible Pitfalls
- if save takes a while can be disruptive to typing
- can cause interface change (as in gmail) leading to unexpected behavior

Best Practice
- make auto-save as transparent as possible
- for email, make it default behavior
- for documents, allow it to be turned on
- catch navigation away from page and offer to save
patterns. inline assistant.

inline assistant.
pattern. inline assistant.

**Use When**
- want to simplify a selection process
- want to remove “bail-out” opportunities
- want to increase conversion rates

**Possible Pitfalls**
- trying to use for complex wizards
- trying to use for installation
- if logic is too complex or dependent on server data, might be sluggish

**Best Practices**
- avoid using inline assistant for branching logic style wizards
- avoid using inline assistant for installation processes
- make full use of preview techniques to clearly indicate each of the small choices the user makes
- make the experience feel lightweight, quick and responsive
patterns. validate.

inline validation. validate then suggest. character counter. live preview
pattern. inline validation.

**Use When**
- want to prevent errors early
- can provide timely feedback in compact form within current form

**Possible Pitfalls**
- sluggish performance

**Best Practices**
- provide feedback as the user types if possible; provide feedback on action button otherwise
- for feedback during typing place feedback in proximity beside (usually to the right) of input
- for feedback after action button, overlay the input area with the feedback; subsequent action button will bring back form if error.

---

**Required information for Google account**

**Your current email address:** someone@somewhere.com

E.g. myname@example.com. This will be your username and sign-in.

**Choose a password:**

*Password strength:* Too short

Minimum of 6 characters in length.
pattern. validate then suggest

**Use When**
- need to aid the user in correcting their errors

**Possible Pitfalls**
- vague messaging

**Best Practices**
- if possible, prevent the user from getting to this point
- clearly denote the fields where issues exist
- make the task if fixing issues seem like 1-2-3 steps
- as the user corrects enable action buttons and remove error indicators
pattern. character counter.

Use When
- input is limited
- can’t intelligently handle long text

Possible Pitfalls
- relying on this when possibly could handle larger input lengths

Best Practices
- update character counter in real time
- character counting text should be written as a complete sentence
- use smaller font, place underneath field

Ask Your Question

1. Enter your question
You can ask 5 more questions today.

You have 110 characters to work with. Add details below.
Pattern: Live Preview

**Use When**
- can provide a visual view of user selection
- can provide the view in real time

**Possible Pitfalls**
- can be very challenging to come up with the appropriate way to combine edit & preview in one view

**Best Practices**
- let hover reveal previews
- provide a preview panel that contains the sum of the choices
- update preview panel in real time, if possible; else have a preview button to show preview
- use narrowing choices pattern where possible
patterns. indicators.

busy indicator. progress indicator. inline status. opacity fade. high contrast.
pattern. busy indicator.

**Use When**
- need to show system is processing
- want to show indication in context

**Possible Pitfalls**
- can be distracting if not necessary

**Best Practices**
- place the busy indication as close to the use input as possible
- use small animated indicator beside input or inside input field
- place the busy indication at the place where the results will appear
- use an overlay with translucency if redirecting attention
- don’t use too many indicators as it will make for a noisy interface
- avoid using indicator if delay is really short
pattern. progress indicator.

Use When
- needing to show upload status

Possible Pitfalls
- non-accurate progress indication

Best Practices
- place the indicators as close to the file names as possible
- finish with “completed” status if displayed in grid... otherwise finish with status and fade out
pattern. inline status.

**Use When**
- needing simple status on page

**Best Practices**
- place the status message in close to where the user is currently interacting or at a visual hotspot (visual heat map)
- highlight the background
- optionally fade out the status message
pattern. opacity fade.

Use When
- need to spotlight a change occurred

Possible Pitfalls
- if too many places, can be distracting

Best Practices
- use in just one or two selected areas
- if more, make sure that there aren't too many updating at once
- if area is large, consider more subtle coloring
- fade out quickly (usually less than one second; but much faster is acceptable)
pattern. black background.

**Use When**
- fade technique does not indicate change well enough

**Possible Pitfalls**
- can be visually noisy

**Best Practices**
- use sparingly
- highlight the background as black (the length of text)
patterns. presentation. transition.

brighten. animate. cross-fade. dim. expand.
fade-in. fade-out. flip. move. self-heal.
collapse. slide.
Transitions

- Brighten. Raise importance
- Cross Fade. One leaves, another enters
- Dim. Lower importance
- Expand. Open in context of others
- Fade In. Good for adds
- Fade Out. Good for deletes
- Flip. Edit configuration
- Animate. Change relationship, tweening
- Self-Healing. After delete or “yank”
- Collapse. Close in context of others
- Slide. Similar to expand but child window
pattern. transitions

**Use When**
- speed up perceived time
- slow down interaction
- show relationships
- suggest symmetry
- show state change

**Possible Pitfalls**
- use the wrong transition can communicate the wrong message

**Best Practices**
- the more rapid the change the more important the event.
- rapid movement is seen as more important than rapid color change.
- movement toward the user is seen as more important than movement away from the user.
- very slow change can be processed without disrupting the user's attention.
- animation can be used to communicate where an object was and now is