Views 2
Designing the user interface

Roy Scholten
hi
Views
Views 2
Views 2
have you heard of it?
have you heard of it?

NOW 12.682% MORE FLEXIBLE
Fields are only meaningful with List view and Table View; they allow you to choose which fields are presented and ordered.

Arguments

This view currently has no arguments defined.

Add Argument

Arguments are parsed directly from the URL. They are not necessary to any given view, but allow flexibility.

Filters

Filter by whether or not the node is published. This is recommended for most Views!
Problem:

How to avoid the humongous scrolling form of death?
ping
This is the default display

Path: test/view/path
Title: Some view, eh?
Menu: Has menu, is default tab, parent is menu item
  Edit menu
Output style: Fields

Fields

Node: title
Book parent: Node: title

Relationships

Book parent

Arguments

Node: book parent
Node: feed selector

Sort criteria

Node: type, ascending
Node: post date, descending

Filters

Node: type is "book", "story", or...
Book parent: Node: nid is 1337
Huh?
Select the keyboard layouts, input methods, and palettes for the input menu.

- **Character Palette**
  - Input type: Palette
  - Script: Unicode

- **Japanese Kana Palette**
  - Input type: Palette
  - Script: Japanese

- **Keyboard Viewer**
  - Input type: Palette
  - Script: Unicode

- **Hangul**
  - Input Method: Korean
  - Options:
    - 2-Set Korean
    - 3-Set Korean
    - 390 Sebulshik
    - GongjinCheong Romaja
    - HNC Romaja

- **Kotoeri**
  - Input Method: Japanese

**Input menu shortcuts**
- Select previous input source: %Space
- Select next input source in menu: %\Space

**Input source options**
- Use one input source in all documents
- Allow a different input source for each document

Show input menu in menu bar
Select the keyboard layouts, input methods, and palettes for the input menu.

<table>
<thead>
<tr>
<th>Input type</th>
<th>Script</th>
</tr>
</thead>
<tbody>
<tr>
<td>Palette</td>
<td>Unicode</td>
</tr>
</tbody>
</table>

- Character Palette
- Japanese Kana Palette
- Keyboard Viewer
- Hangul
- 2-Set Korean
- 3-Set Korean
- 390 Sebulshik
- GongginaCheong Romaja
- HNC Romaja
- Kotoeri

Input menu shortcuts

Select previous input source:
Select next input source in menu:

Input source options

- Use one input source in all documents
- Allow a different input source for each document
- Show input menu in menu bar

For each modifier key listed below, choose the action you want it to perform from the pop-up menu.

- Caps Lock (⇪) Key: Caps Lock
- Control (^) Key: Control
- Option (⌥) Key: Option
- Command (⌘) Key: Command

When this option is selected, press the F1 key to use the F1-F12 keys to control hardware features.

- Illuminate keyboard in low light conditions

Turn off when computer is not used for:

- 5 secs
- 10 secs
- 30 secs
- 1 min
- 5 mins
- Never

Modifier Keys...
round 1: groups.drupal.org
### Page 1: Node

**This is the default display**

- **Path:** test/view/path
- **Menu:** Has menu, is default if content is menu item
- **Output style:**
  - Fields
  - Settings

#### Fields

- **Node: title**
- **Book parent: Node: title**

#### Relationships

**Sort criteria helpext, ladida.**

- **Arguments:**
  - Node: type, ascending
- **Filters:**
  - Node: post date, descending
- **Sort criteria:**
  - Book

#### Arguments

**Delete**

**Add field**
Title:

The unique identifier of the view; it is only important for overridden views and views that modules or themes will need to use.

Access:

- admin
- Adrian
- anonymous user
- authenticated user
- Dan

Only the checked roles will be able to see this view in any form; if no roles are checked, access will not be restricted.

Description:

A description of the view for the admin list.
“Guys, you're all redesigning Views1 here…”
ping
“I haven't asked Merlin thoroughly enough about his starting points really.”

“I would say grokking Merlin's intentions is step one. Otherwise we are designing the UI in a fog.”
What do I want to display?
How do I want to display it?
round 2: Sentences
(or: “What is really going on here?”)
horizontal for the formats you mean?alpritt: If there is not really a limit to how many multiples you can have, yes.yoroy: true.alpritt: Though tabbed itis hard for me to have that issue too.yoroy: yeah, what are your feelings on vertically oriented tabs like here: http://groups.drupal.org/files/Views-UI-bottomdrag.html?alpritt: I/do have that feeling of wanting to have 2 slight variations of the block2.yoroy: yes.alpritt: It is a possibility.alpritt: What is the for or only using one view?yoroy: Yeah, we should ask the evil genius do you recognize the feeling of wanting to have 2 slight variations of the view?alpritt: Sure.alpritt: But they are being treated as completely different views by the looks of things.yoroy: Building a whole other block just that little variation is a bit awkward to me.alpritt: Is that what you are doing.alpritt: Except for the name, description and basic settings.yoroy: remember that relationships, sorts, arguments and filters can be different for all output formats.alpritt: anyway: it's just the fields part what is the forks so to speak.yoroy: the last 2 screens in views.drupal.org/files/views-ui-3steps.png are not correct.alpritt: This is not how we have multiples of them. Additionally, fields and arguments will be different for each view, so different output types can have their own sets of arguments**.yoroy: we should check, I think this has been discussed. The arguments bit.yoroy: http://groups.drupal.org/node/6268#comment-17931.yoroy: Is this comment from a different view of the angydonut posting right?alpritt: That's right.alpritt: The output section last.alpritt: or would it? alpritt thinks that his mockup is contrary to what he said there.alpritt: I think I will stick with the output section last.alpritt: or would it? alpritt thinks that his mockup is contrary to what he said there.alpritt: I think I will stick with the output section last.alpritt: or would it? alpritt thinks that his mockup is contrary to what he said there.alpritt: I think I will stick with the output section last now.alpritt: As in you process everything, making certain nodes etc, and then you output the result in different ways.alpritt: Like the rower works.alpritt: Like .tpl.php files work.yoroy: so, yoroy: first: define your list of items.yoroy: then: how would you like to present them.alpritt: Yes.yoroy: sounds good.alpritt: Is it backwards at the moment?yoroy: yes.current setup after naming your view is the "page" and "block" fields set by your views. view 1 that is.yoroy: below that are the optional fields.yoroy: So you start with a big list of everything.yoroy: filter it down.yoroy: ooh, that's a nice bottleneck you put in that that is a bit backwards yeah.alpritt: I can't decide if that is a good thing or not.alpritt: Well.alpritt: It would still work like that though.yoroy: yes, one person might come in with the mindset of "block" alpritt: You would still start with everything.alpritt: So section 1: what do I want to display?alpritt: So you can filter it.alpritt: instead of...alpritt: 1: How do it?alpritt: 2: What do I want to display?alpritt: They are logical.yoroy: yes.alpritt: In fact.alpritt: Why not do questions.alpritt: Right now we have sections that make Basic information.alpritt: Page alpritt: Block alpritt: Field Arguments alpritt: Filters alpritt: Exposed Filtersyoroy: Right now we have "fork" only in the second part, keeping it "generic" in the first.yoroy: yep.alpritt: Rather than fork first. Makes no sense.alpritt: These sections don't make any sense unless you know what they mean.alpritt: You have to learn it.alpritt: Two sections.yoroy: what and how.alpritt: What do you want to
What do you want a list of,
What do you want a list of,

Where do you want to show it,
What do you want a list of,
Where do you want to show it,
How to present it, using which
What do you want a list of,
Where do you want to show it,
How to present it, using which Format for each item?
What do you want a list of,
Where do you want to show it,
How to present it, using which Format for each item?
What do you want a list of,
Create new view

Enter the **name** of the view

**What do you want to display?**

- **Nodes**
- **Block**
- **Embedded**

Choose your default display. You can add more later.

Choose how you want to present this view.

- **Table**
- **???”

[Create view]
What

Where do you want to show it,
page
default view

block

add new view

path

× use default settings

Fields

Filters

Arguments

Sort
Displays

Page *(default)*

Block
RSS Feed

Add Display:
- Page
- Block
- Embed
- Contrib module

Add

Choose the kind of list for this display.

- List
- Table
- Plain

Configure how each list-item is presented

- Teaser
- Full posts
- Choose Fields...

Filters
Arguments
Sort
What do you want a list of,
Where do you want to show it,
How to present it, using which Format for each item?
“… a system where the base edit page is basically not editable, but gives you a summary of everything on the view and then leads you to separate pages where you can edit these items…”
Basic information

**View:** views_test2
**Type:** Node
A view being used to test some handlers.

Edit this

Page (Default)

Display information goes here.

Edit this

relationships

None defined.

Edit this

fields

Node > Title  
(Additional information here)
Node > Post date  
(Additional information here)
Node > Body  
(Additional information here)
Node > Teaser  
(Additional information here)
Node > Published  
(Additional information here)
Node > Promoted to front page  
(Additional information here)
User > Name  
(Additional information here)

Edit this

arguments

Node > Type  
(Additional information here)

Edit this
Summaries
A list of nodes displayed in...

<table>
<thead>
<tr>
<th>Pages</th>
<th>Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>page_1</td>
<td>block_1</td>
</tr>
<tr>
<td>page_2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>example.com/path/to/view</th>
<th>arguments</th>
<th>Unformatted list of</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>/</td>
<td>node type</td>
</tr>
<tr>
<td>Footer</td>
<td>x</td>
<td>relationships</td>
</tr>
<tr>
<td>Empty text</td>
<td>/</td>
<td>none defined</td>
</tr>
<tr>
<td>Page</td>
<td>10 nodes per page</td>
<td>filters</td>
</tr>
<tr>
<td></td>
<td></td>
<td>node type</td>
</tr>
</tbody>
</table>

We can keep some consistency here so we aid habituation for advanced users. But also it should be clear for new users.

This is some header text
the_name_of_my_view  
is a node list

this is what I can say about the view

---

### general settings

Filter by whether or not the node is published. This is recommended for most Views!

Filter by node type. Include or exclude nodes of the selected types.

---

### Displays

Page name: example.com/path/to/view

Arguments: not-defined

Relationship: not-defined

Filters:
- Node: Published
- Node: type
- Node: updated

Sort Criteria: Created time

Title: Page Name

path: * path/to/view

---

Update Display
a month later…
View **views_test2**, displaying items of type **Node**.

<table>
<thead>
<tr>
<th>Page (Default)</th>
<th>Basic display info goes here (TBD)</th>
<th>Relationship</th>
<th>Add · Rearrange</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>None defined.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Add · Rearrange</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Node: Type</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Add · Rearrange</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>None defined.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Add · Rearrange</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node: Title</td>
<td></td>
</tr>
<tr>
<td>Node: Post date</td>
<td></td>
</tr>
<tr>
<td>Node: Body</td>
<td></td>
</tr>
<tr>
<td>Node: Teaser</td>
<td></td>
</tr>
<tr>
<td>Node: Published</td>
<td></td>
</tr>
<tr>
<td>Node: Promoted to front page</td>
<td></td>
</tr>
<tr>
<td>User: Name</td>
<td></td>
</tr>
<tr>
<td>Sort</td>
<td>Add · Rearrange</td>
</tr>
<tr>
<td>Node: Post date</td>
<td></td>
</tr>
</tbody>
</table>

Click on an item to edit that item's details.
Edit view "fugazi"

View fugazi, displaying items of type Node.

<table>
<thead>
<tr>
<th>Defaults</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Page</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Block</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Block</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Add display</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**View settings**
- Tag: fugazi

**Basic settings**
- Name: Defaults
- Title: Fugazi
- Style: Unformatted
- Row style: Node
- Use pager: No
- Items to display: 10
- Access: Unrestricted
- Header: Filtered HTML
- Footer: Filtered HTML
- Empty text: Filtered HTML

**Relationships**
- None defined

**Arguments**
- Node: Created year + month
- Style: Default

**Sort criteria**
- None defined

**Filters**
- Node: Published = Published
- Node: Type in Page, Story

**Defaults: Access restrictions**

<table>
<thead>
<tr>
<th>Type</th>
<th>If by role:</th>
<th>If by perm:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unrestricted</td>
<td></td>
<td>administer blocks</td>
</tr>
</tbody>
</table>

- By role:
  - anonymous user
  - authenticated user

- By perm:
  - Only users with the selected permission flag will be able to access this display.

- Update  Cancel
Good?
- vertical tabs
- summaries
- compact
Bad?

- overwhelming
- inconsistent with core
- abstract terminology
Let's do this more often...
Tips

Ask for input
Tips

Design before coding
Tips

Wireframes
Tips

Usability Testing Suite
(d.o/project/uts)
What's next?

Let's do this more often
discuss...
Thank you!
(encore)
View fugazi, displaying items of type Node.

Block 1: Display the view as a block.

Basic settings:
- Name: Block 1
- Title: Fugazi 2
- Style: Unformattted
- Show style: Fields
- Use MAX: Yes
- Use pager: Yes
- Items per page: 5

Access: Unrestricted
- Link display: Admin page
- Header: Filtered HTML
- Footer: PHP code
- Empty text: Filtered HTML
- Theme: Information

Block settings:
- Admin: Fugazi block 1

Relationships:
- None defined

Arguments:
- Global: Null

Filters:
- Node: Published exposed
- Node: Type exposed
- User: Current exposed
- Node: Comment status = Read/Write
- Taxonomy: Term exposed

1) All displays on this view.
2) Indicates currently edited display.
3) Clicking item displays the form here.
4) The currently edited item is highlighted.
5) Changed items are highlighted.
6) Click this button to change whether this item uses default values or overridden values.
7) Lighter color indicates item is using values from default display.

Block 1: Items per page

Status: using overridden values.

Offset:
- 0
  The number of items to skip. For example, if this field is 3, the first 3 items will be skipped and not displayed. Offset can not be used if items to display is 0; instead use a very large number there.

Update  Cancel  This cancel button will cancel changes to just this item.

This cancel button will cancel all changes to the view.
Save is disabled while a form is open to prevent clicking on the wrong update button.
Views 2
Designing the user interface

Roy Scholten
http://www.yoroy.com