Specifying Feedback

- Submitting new feedback
  - Before submitting
  - Submission process
- Participating in the feedback process
  - Feedback issue state machine
  - Additional elements
  - Understanding management actions
  - Interacting with issues
- Searching and monitoring issues
  - Pre-defined searches
  - Defining/constraining your own search
  - Favorite searches and other searches
  - Bulk changes

Tutorials

- How to Bulk Edit-HL7 JIRA
- How to Comment-HL7 JIRA
- How to Create An Issue-HL7 JIRA
- How To Edit And Withdraw-HL7 JIRA
- How To Perform Record Resolution-HL7 JIRA
- How To Perform Triage-HL7 JIRA

Specification Feedback projects are the official mechanism for providing feedback about any HL7 specification. Each product family has its own feedback project and any specification without a product family will use a single feedback project. The ones currently defined are:

- CDA
- FHIR
- V2
- Other - for all specifications at HL7 that do not have a product management group - for example, CIMI, V3, Domain Analysis Models, EHRs, Functional Model and profiles, etc

These projects are used for submitting feedback related to HL7's Ballot Process as well as all suggestions/issues/concerns of implementers and other users of HL7 specifications. The mechanism becomes available once a specification has progressed to ballot or when the associated work group asks for formal feedback to be enabled.

This page covers three main topics:

- Submitting new feedback
- Participating in the feedback process
- Searching and monitoring issues

This page is aimed at those involved in submitting feedback. Additional documentation is available for co-chairs and others who manage issue resolution or who act as administrators for the feedback process. For those familiar with the FHIR gForge submission tool which Jira replaces, there's also a page that discusses changes from gForge.

This page provides general guidance that applies to specification feedback regardless of which HL7 product family it belongs to. Additional family-specific guidance is also available (read this page first):

- CDA Specification Feedback
- FHIR Specification Feedback
- V2 Specification Feedback
- Other Specification Feedback

Submitting new feedback

An essential part of the standards development process is receiving and managing feedback from the community. Without feedback, HL7 can't know if its specifications are appropriate or useful to their target community. We welcome feedback from anyone and everyone who is willing to be constructive. Feedback might be suggesting a feature, pointing out a place where a specification is unclear or over-restrictive, identifying a spelling or grammar issue or suggesting that an entire area needs to be rethought. HL7 makes a special effort to solicit feedback using our ballot process, however feedback can be submitted at any time by anyone - even if they're not an HL7 member. (That said, we strongly encourage membership, both to support the standards community and because it allows you to give your feedback greater weight.)

Before submitting

In order to submit feedback, you must register as a user on HL7's Confluence and Jira systems. Individuals are encouraged to submit feedback themselves. Submitters will automatically be notified as a submission is commented on or achieves milestones within the review process. They may also be asked questions about and/or invited to calls to discuss their feedback. Therefore, it's best for feedback to be submitted directly by the individual directly impacted by or having direct knowledge of the issue being submitted.

Before submitting feedback, users are encouraged to browse through previously submitted feedback to see if the topic has already been discussed and, if so, what discussion has already taken place. It may also be helpful to search the appropriate chat forum. Duplicate requests will be closed without discussion unless they raise new points for consideration.

Remember that all issues and comments submitted in Feedback issues are subject to the HL7 Acceptable Use Policy.
Submission process

Feedback can be initiated by clicking on any of the links above, by clicking on the “Create” button from http://jira.hl7.org, or in some cases, by using a link within an HL7-published specification.

General guidance

When submitting an issue, you will be prompted with two fields that determine the "kind" of issue being reported - and which in turn determine what fields are available to describe the issue and what the validation rules are for submitting the issue.

<table>
<thead>
<tr>
<th>Issue Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change Request</td>
<td>This is the default choice. It represents a request for change to the referenced specification that is more significant than a mere technical correction. It will be resolved with a vote by the appropriate work group. (For those familiar with HL7's old ballot commenting scheme, this would be used for items flagged as NEG, A-C or A-S.)</td>
</tr>
<tr>
<td>Technical Correction</td>
<td>This is a minor change to the specification that does not impact the behavior of conformant applications. It includes corrections to spelling and grammar, errors with propagating name changes of elements within the specification and formatting issues. These sorts of issues will be approved automatically once verified. A facilitator with appropriate access can mark as 'Will Fix' (option under 'Workflow' drop down). In the old balloting system, this would correspond to A-T.</td>
</tr>
<tr>
<td>Question</td>
<td>This sort of feedback doesn't anticipate a change to the specification, but instead it seeks a clarification on some aspect of it. In most cases, questions should be raised on <a href="http://chat.hl7.org">http://chat.hl7.org</a>, <a href="http://chat.fhir.org">http://chat.fhir.org</a>, or one of HL7's listservs. These mechanisms often result in a response in minutes or hours, while questions raised as a feedback issue might not be responded to for weeks or months. This option for feedback remains primarily to satisfy ANSI ballot requirements. In the old balloting system, this would correspond to A-Q. If a question requires a change, it should be moved to a 'Change Request'.</td>
</tr>
<tr>
<td>Comment</td>
<td>Comments are commentary about the specification that is not expected to result in a change to the specification – or any response at all. For example, “Great job on the specification!” It too exists to support ANSI ballot requirements. To save on administrative effort, commenters are encouraged to provide such feedback by means other than Jira. In the old balloting system, this would correspond to A-A.</td>
</tr>
</tbody>
</table>

The Issue Type can be important when making submissions as part of an HL7 ballot. Of the various types, only a “Change Request” can be tied to a negative vote. All other types are restricted to “affirmative”, meaning that the feedback item cannot block publication of a specification.

It is possible for the Project and/or Issue Type to be changed after the issue has been submitted – either because it wasn’t categorized correctly or based on the decision of the work group. For example, an issue submitted as a “Question” might be updated to a “Change Request” if the work group decides that the specification should be changed to provide further clarity.

Specification Issue Tabs
The issue submission tab is split into a Default and Advanced tab. The advanced tab is intended for HL7 insiders who are already familiar with work groups, categorization conventions and other practices used as part of issue management. This tab can be ignored if you aren’t sure what to fill in – someone at HL7 will do so for you when relevant.

Each tab contains several fields. Only those marked with an asterisk are mandatory, though there are conditional requirements that mean at least some of the location-related fields must also be filled in. While there is no need to fill in all of the available fields, submitters are encouraged to fill in as many as they can. The more complete the feedback item is, the easier the committee will be able to evaluate it.

**Specification**

The next drop-down is the specification. A specification must always be identified. Feedback not related to a specific HL7 specification (e.g. related to the HL7 website, to staff, to meetings, software products, etc.) must be made elsewhere.

The list of available specifications is driven by the choice of feedback Project. Each project contains specifications only for that one project. If you can’t find the specification desired, try looking under a different project. If you still can’t find the specification, reach out on a list serve or chat forum or contact the webmaster. If the Project field changes, the Specification element will automatically be wiped.

**Raised in Version**

Issues should generally only be raised against the current version of the specification (and this field will automatically default to that once the issue has been created). Previous versions of a specification can only be updated in the event of obvious errors, and then only in ways that would not break implementations of that version. Enhancements should always be proposed based on the most recent version of the specification made available.

**Summary**

This field must be filled with a short sentence describing the issue. This is what everyone will see in the “search” view when looking at a list of issues, so try to ensure that the description is useful and precise. However, it should also be relatively short (generally less than 100 characters or so). Additional information can go in the Description element further down.

**Location details**

The next few fields vary by Project/product family. Each family has its own approach to publication and therefore its own way of identifying what part(s) of the specification are relevant to the feedback. Each specification will enforce that the location is identified in some manner (or that a positive assertion is made that no specific location is relevant). Guidance on the location fields can be found in the product-specific feedback documentation:

- CDA Specification Feedback
- FHIR Specification Feedback
- V2 Specification Feedback
- Other Specification Feedback
During the maintenance process, the set of associated artifacts might be corrected, supplemented or otherwise adjusted during the triage process or
as part of work group review.

Description
This field can be omitted if the issue is fully described by the summary. However, in most cases, additional detail will be needed. That detail goes
here. This might include a quote of the current text, proposed replacement text, rationale for a change, links to additional examples or any other
information that might be relevant to the work group evaluating the change.

This field allows formatting, including lists and tables if that is helpful to better present the information.

Attachment
Supporting information such as screen-shots, example instances or other relevant information to support an issue can be attached here.

Request in-person
HL7 processes allow a submitter to request that they be present when a submitted issue is discussed (be that at a conference call or face-to-face at a
working group meeting). The specification feedback extends that capability to identify additional people who should be present as part of the
discussion. This element can be used to list one or more Jira users who the submitter feels should be involved in any decisions relating to the
outcome of the issue.

Any individuals specified here will form the initial set of “in person” users. However, users can add and remove themselves from the list if they wish
and managers can add additional individuals.

Note that a request for “in-person” consultation is not a guarantee. Work groups will make their best effort to include those who have requested in-
person in discussions. However, if after making reasonable efforts at accommodation, the responsible workgroup is free to make a decision even if
not all of the identified users are able to be present.

Note that selecting “in person” imposes an administrative hardship on the work group and can slow down the resolution process. As a result, this
feature should not be used except for feedback items the submitter considers particularly important and where they feel their presence might help to
inform or sway opinion.

The remaining fields are found on the “Advanced” tab. They will typically be filled in as part of the triage and issue management process. They
should only be filled in on submission by users familiar with HL7 processes and who are confident in their ability to pick appropriate values.

Work Group
This indicates the work group (or occasionally other HL7 group) that is currently responsible for reconciliation of the issue. It will be based on the
selected specification, the relevant location(s) within the specification and/or the nature of the of the issue raised.
In some cases, responsibility for an issue may be transferred between work groups – either due to an incorrect initial assignment or because as the understanding of the issue evolves, an alternate view-point is deemed necessary, or the type of change needed is determined to be the responsibility of another group.

**Priority**

All issues start with a “medium” priority. However, work groups can alter the priority to allow them to focus review on issues they deem of greater importance – for example, those tied to negative votes, those likely to require more discussion, those likely to result in substantive change, etc.

Priority should not be asserted by anyone not part of HL7 governance processes or representing the work group currently assigned to the issue.

**Pre Applied**

In some cases, issues may be raised by HL7 stakeholders who have commit rights to the underlying specification. These users might choose to pre-apply a proposed fix in order to help the reviewing work group better understand the nature of the proposed change. In some cases, a change might be applied after submission but before work group review for the same reason.

If a change is rejected or altered, the committed change will need to be reverted or adjusted accordingly.

**Grouping**

This field allows one or more categories to be assigned to an item. These might include associating the item with a particular project, linking a bunch of related issues with a shared category, identifying issues that are ready for vote or other review, etc. Categories can be defined at will and all category values currently in use will be selectable by type-ahead.

Note: These categories span all Projects/product families. As a result, care should be taken to make them general purpose when possible. Tags that are no longer relevant should be removed from issues if they are unlikely to be needed for searching.

**Scheduling**

This field works similarly to the Grouping tag. However, the focus here is on scheduling the item. It could contain the date of a particular call, or something more generic such as “next call”, “WGM discussion” or even a particular WGM quarter. Again, values are shared. Non-general-purpose tags should be removed when they are no longer relevant.

Changes to this element will result in notifications being sent to all *Request in-person* users.

Note that scheduling an item represents an intention, not a commitment. Issues can often be taken up before or after they are scheduled, though best efforts will be made to ensure participation of “in person” requesting individuals.

**Related Issues**

Some issues may be related to pre-existing issues. For example, a feedback item might be a request to re-open a previously resolved issue, might be proposing a similar change to another issue (but to a different area of the specification), might be proposing a change that follows on from another issue, etc. This field allows links to be asserted to any other issues in the HL7 Jira site and provides for easy cross-referencing between the linked issues.

TODO: Verify that asserting a link results in a bi-directional link

**Participating in the feedback process**

Submission of a new feedback item initiates a process of review by HL7 members. This section describes the cycle of that review and indicates how submitters can engage in the review process at different stages.

**Feedback Issue state machine**

All specification feedback issues are governed by a shared state machine:
All issues will start out as "Submitted" (Blue) and will eventually end in one of the bottom states (green). The following table provides an explanation of each of the states:

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Submitted</td>
<td>This is the initial status for all new feedback items. It indicates that the item needs to be reviewed to verify that it’s assigned to the right work group, is appropriately categorized, etc. Work groups generally don’t look at items in the 'submitted' status. Specific individuals at HL7 are responsible for checking the newly submitted items, correcting them if necessary and then transitioning them to Triaged. (Though in some cases the manager reviewing the proposal might automatically transmit it to a final status.) In most cases, issues shouldn't remain in this state for more than a few business days, though in some cases it might take a week or two.</td>
</tr>
<tr>
<td>Triaged</td>
<td>This is an issue that is ready for work groups to evaluate and make a decision on.</td>
</tr>
<tr>
<td>Waiting-for-Input</td>
<td>This indicates that the issue cannot be resolved until more information is provided. The individual who needs to provide further information is identified in the &quot;assignee&quot; field. That individual will receive weekly emails until the requested information is provided.</td>
</tr>
<tr>
<td>Resolved - No change</td>
<td>No change to the specification will be made based on this issue. This could be because the submitter withdrew the item, the work group decided not to make any changes based on the issue, changes have been made based on other (non-duplicate) issues that render the change unnecessary, or the issue wasn't requesting a change in the first place (e.g. it was a question) and the issue is now resolved (e.g. it was a question).</td>
</tr>
<tr>
<td>Resolved - Change Required</td>
<td>A change will be made to the specification based on this issue (though not necessarily what was proposed).</td>
</tr>
<tr>
<td>Duplicate</td>
<td>This issue has been determined to be substantively the same as another issue within the same project. This issue will not be resolved - instead the resolution of the linked duplicate issue should be consulted instead.</td>
</tr>
</tbody>
</table>
Deferred
No action will be taken against this issue for this release, but it will be returned to "triaged"

Applied
The agreed change has been made to a draft version of the specification, but not yet included in an official release

Published
The agreed change has been included in a published version of the specification

Additional elements
Once an issue has been created, a few more fields become available that you may see when looking at an issue

Understanding management actions

Interacting with issues
Once an issue has been submitted, what you can do with that issue depends on what permissions you have as an HL7 Jira user.

- Everyone - including those who aren't registered users - can read and search against issues.
- Registered users have an additional set of capabilities, described below.
- Other pages describe actions available for managers and administrators.

Workflow actions
Many of the actions that can be taken on an issue fall under the category of "workflow" issues. These are actions that involve pushing a button and potentially filling in information on a separate screen. Depending on the active user's permissions, the issue's status and other factors, the set of available actions will vary. Jira will only display 3 actions "on-screen". If more actions are possible, it will display a Workflow button that you can click to see additional actions. (So if something you want to do isn't on the screen, check to see if it's hiding under Workflow.)

Editing an issue
Regular submitters (those without manager or administration privileges) can edit issues they've created, but can only do this while the status remains Submitted. Once the issue has changed to Triaged or any other status, it can no longer be edited (though you can add comments - see further down). The fields available for editing are the summary, description, location fields, attachments and linked issues. The project, issue type and specification cannot be changed. If those are incorrect, the issue can be closed (see below) and a new one created.

Note: Editing is only allowed if there are no links to the issue. If it's been voted on, is the target of a duplicate reference or any other type of Jira reference, it will no longer be editable.

Closing an issue
Submitters who have opened an issue are free to "close" the issue so long as it remains in an un-resolved state (status is Submitted, Triaged or Waiting for Input). Doing so will automatically transition the issue to Resolved - No Change Required with a resolution indicating that it's been withdrawn by the submitter. Note that this can only be done by the submitter of the issue, not by anyone else (even if nominally acting on the submitters behalf).

Note: Like editing, closing is only allowed if there are no links to the issue. If it's been voted on, is the target of a duplicate reference or any other type of Jira reference, it will no longer be possible for a regular submitter to close their own issue.

Requesting/Un-requesting "in-person"
Any registered HL7 Jira user can add or remove themselves from the "In-person" list associated with an open issue by pressing the "In Person" or "Not In Person" buttons (the former is only available if the user is not currently on the list, the latter is only available if they are). Work groups will do their best to ensure that individuals noted as "in-person" are aware of when the issue will be discussed and will try to arrange times such that those individuals can be present.

Managing votes on an issue
Various vote-related options will be available if an issue is deemed related to a ballot that the user is signed up for that is open and/or if the user has submitted a vote on the item. Possibilities include Vote, Un-vote, Retract and Withdraw. More details about these options and the situations in which they are available can be found on the Jira Ballot Process page.

Other actions
These are general actions that are available for most types of Jira trackers and have specific controls on the issue to manage them.

Watching/Un-watching an issue
Users can add themselves as "watchers" of an issue. Watching an issue means that if the issue is changed in some way (edited, comments added, transitioned to a different state, etc.), the watching user will receive an email notifying them of what happened. The submitter of an issue is automatically added as a watcher.
To add yourself or remove yourself as a watcher, click on the link beside the Watchers count or choose the option from the More menu. You can see who's watching an issue by either clicking on the "count" of watches or selecting More, Watchers from the issue menu.

Commenting on an issue

All registered users can post comments against existing issues, regardless of who created the issue. These comments might provide additional use-cases, indicate agreement or disagreement, suggest alternatives or provide any other commentary intended to be useful to the work group in making their decision about how best to resolve the issue. Commenting is only allowed on "open" issues. Once an issue is resolved, only managers and administrators can make further comments. If you have feedback on an issue that's already resolved, that needs to be provided by reaching out on the appropriate chat forum or list server or by creating a new issue (ideally with a link to the one your feedback relates to).

To make a new comment, just press the button.

Looking at the history of an issue

If you're interested in seeing what's happened to an issue over time you can look at the History, Activity or Transitions tabs near the bottom of the issue. They each show different views of what's happened.

History provides a complete chronological list of changes to fields in the issue.

Activity provides a recent list of changes, comments, links to and from the issue

Transitions provides a list of the state changes the issue has undergone

Additional issue fields

After submission, additional fields come into play. Two of them (Watchers and Comments are discussed above). The additional fields are:

Assignee

For 'open' issues, this indicates who the "Waiting for input" individual is. (If the issue is open and there's an assignee, the status will always be "Waiting for input". For closed issues, assignee should only be present if the status is "Resolved - Change Required" and in that case, it indicates who is expected to perform the change. If you're wanting to know when an issue will be progressed, following up with the Assignee isn't a bad idea, but keep in mind that they're a volunteer.

Change Category

This is captured if a decision is made to change the specification. It categorizes the change as being either a correction, clarification or enhancement. These categories are used for reporting as we examine how different parts of a specification are evolving over time.

Change Impact

This field tracks whether a change to the specification is considered 'substantive' (which has impact on the HL7 Balloting process) and whether a change is a 'breaking' change, which is generally of interest to the implementer community, and which also has implications for whether a change is allowed or not, depending on the level of standardization of that portion of the specification.

Outstanding Negatives

This indicates whether there are any outstanding negative votes associated with the issue, and if so, what the highest 'level' of outstanding negatives is. (Normative is higher than STU which is higher than informative which is higher than draft). It's a calculated element based on associated ballot submissions and withdrawal status.

Reporter

This is the individual who originally created the issue.

Resolution

This categorizes the decision made by the work group about the feedback issue raised. Every closed issue will have a resolution. The possible resolutions are:

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Issue Type</th>
<th>Resulting Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Considered - No action required</td>
<td>Comment</td>
<td>Resolved - No change</td>
<td>Comments never require any action. This resolution means that the comment has been reviewed and that no further action will be taken.</td>
</tr>
<tr>
<td>Considered - Question answered</td>
<td>Question</td>
<td>Resolved - No change</td>
<td>The Resolution Description will contain the answer to the question.</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>---------</td>
<td>---------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>Considered for Future Use</td>
<td>Change Request or Technical Correction</td>
<td>Deferred</td>
<td>The work group reviewed the proposal and don’t wish to make a change in the current release of the specification, but will consider making a change in a future version of the specification. The Resolution Description may indicate the reason for waiting.</td>
</tr>
<tr>
<td>Persuasive</td>
<td>Change Request or Technical Correction</td>
<td>Resolved - Change required</td>
<td>The proposed change will be made more or less as proposed.</td>
</tr>
<tr>
<td>Persuasive with Modification</td>
<td>Change Request</td>
<td>Resolved - Change required</td>
<td>An A change will be made to the specification that addresses the issue identified, but it is noticeably different than the one proposed. The Resolution Description will indicate what change will actually be performed.</td>
</tr>
<tr>
<td>Not Persuasive</td>
<td>Change Request or Technical Correction</td>
<td>Resolved - No change</td>
<td>No change will be made to the specification. The Resolution Description will indicate why. This might be because the work group disagrees with the submission or because the issue isn’t one that can be addressed within the specification.</td>
</tr>
<tr>
<td>Not Persuasive with Modification</td>
<td>Change Request</td>
<td>Resolved - Change required</td>
<td>The work group disagrees with the issue raised and will not make the requested change, but will make some sort of change to the specification - generally to provide more clarification. The Resolution Description will indicate what change will actually be performed.</td>
</tr>
<tr>
<td>Duplicate</td>
<td>Any</td>
<td>Duplicate</td>
<td>The issue is essentially identical to another issue in the same project. The resolution will be captured on the linked duplicate issue.</td>
</tr>
<tr>
<td>Retracted</td>
<td>Any</td>
<td>Resolved - No change</td>
<td>The submitter has closed their own issue prior to resolution.</td>
</tr>
</tbody>
</table>

**Resolution Description**

This provides additional detail about the resolution. This must be specified for all resolutions with the exception of "Persuasive", "Considered for future use" and "Considered - No action required", "Duplicate" and "Retracted".

**Resolution Vote and Vote Date**

This provides a formal record of the work group decision on the issue and is required to align with HL7 policies. It will be present on all closed Change Requests provided that the resolution is not "Duplicate" and the item has not been withdrawn.

**Applied For Version**

This indicates the release the change is expected to be (or has been) applied to. Generally this will be the forthcoming release, but in some cases, applying a change might be bumped to a future release and (on very rare occasions), a change might be applied to a previous release.

**Searching and monitoring issues**

There are many reasons for searching through the set of Specification Feedback issues. For example:

- Checking to see if a proposed change has already been discussed
- Checking what proposals are currently being considered for a particular specification/artifact/area
- Checking what proposals have been approved but not yet applied
- Checking what proposals require review by a specified work group
- Checking the rationale behind a change that’s been applied to a specification (which requires finding the issue that trigged the change)

To search through issues, select Issues, Search for Issues (or Current Search) from the main Jira menu. From the search screen, you can use one of the standard pre-defined searches, open a search you’ve previously saved to your favorites, search for an existing search created by someone else or define your own custom search.

**Pre-defined searches**

Jira provides a set of pre-defined searches that apply across all HL7 Jira projects. This section will describe what they each do and will indicate which ones are likely to be useful/relevant when working with specification feedback items. Several of these have limited usefulness, but unfortunately Jira does not yet provide a mechanism for hiding them. Some of these might provide a useful starting point to further constrain by adding additional filters as described below in "defining your own custom search".

Note: The pre-defined searches marked with "*" have been overridden from default Jira behavior.
My open issues*  
This search identifies all issues assigned to the current user that do not yet have a final status. It is not constrained to specification feedback issues, so issues in terminology projects, action-item projects and others will also appear in this list. The list of items will show highest priority items first and will then be sorted so that issues that have been updated more recently appear closer to the top.

For specification feedback items, issues appearing in this list will either be one of two statuses:

- *Waiting for Input* issues will have a request for information in one of the issue comments. They can be closed by clicking on Provide Requested Input and providing the information requested in the comment field on the issue transition screen.
- *Resolved - Change required* are waiting for the requested change to be applied (and potentially for any pre-applied changes to be rolled back if the requested change differs from the pre-applied change). Once the change is committed, the issue can be closed.

In both of the above situations, users with manager privileges and above can also assign the issue to other users.

Reported by me*  
This lists all issues that were created by the current user, including non-specification feedback issues. (Ballot submissions are excluded from this list as they aren’t really “issues” and aren’t technically raised by the user.)

All issues  
This provides an unfiltered list of all issues in the Jira system. It’s a quick way to start with a blank set of filters from which you can further constrain to the set of issues of interest.

Open issues  
This includes all issues that are not yet in a ‘final’ status, such as “Published” or “Resolved - No changed”. It includes issues from all projects, not just specification feedback items.

Done issues  
This lists all issues that are in a ‘final’ status. The combination of the Open issues and the Done issues will give the same count as shown in All issues.

Viewed recently  
This shows what issues the user has recently created and/or opened. It’s helpful if you remember looking at an issue but forget what its identifier was.

Created recently  
This shows all issues that have been created by any user across all projects in the past week. Typically you’ll want to add additional filters to narrow down the project or possibly specification

Resolved recently  
This shows all issues that have had their Resolution element filled in within the past week. Note that, for Specification Feedback items, the resolution might be filled in as a tentative resolution prior to final vote. Such items shouldn’t truly be considered ‘resolved’, but they will show up in this query.

Updated recently  
This includes all issues that have been created, changed or commented on within the past week across all projects. Again, you’ll probably want to add additional filters to narrow down the project or other characteristics.

Defining/constraining your own search

While the standard searches are sometimes useful, most of the time you’ll want to define your own search - either by starting from scratch or using a standard or saved search as a starting point. This allows you to narrow down the listed issues to those relevant to a particular product family (filter by project), those tied to a particular specification, those submitted by a particular user, those dealing with a particular artifact, etc.

The search dialog provides a "Basic" view and an "Advanced" view, toggled by using the link towards the right on the search bar. The "Basic" view allows filtering on the most commonly used fields and also allows the selection of other fields using the "More" drop-down. The "Advanced" view allows the full power of Jira’s query language (JQL) and allows grouping of conditions using brackets, inclusion of OR as well as AND and a number of additional advanced filtering capabilities. Additional guidance on search can be found in the Jira help by clicking here or by clicking on the ? icon in the Advanced search window.

Some fields that may be useful to search on (e.g. Related Artifact(s), Related Page(s), etc.) may not be selectable unless your search explicitly filters on Issue Type. That's because these fields are constrained to only be enabled for certain issue types. If you filter and select all four issue types (Proposal, Technical Correct, Question and Comment), you'll avoid constraining out any issue types and will be able to filter by the restricted fields.

Examples

<table>
<thead>
<tr>
<th>Description</th>
<th>JQL</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>All open trackers related to the Patient resource that I’m not watching</td>
<td><code>project = FHIR AND watcher != currentUser() AND Specification = &quot;FHIR Core (FHIR) [FHIR-core]&quot; AND &quot;Related Artifact(s)&quot; =&quot;Patient [Patient]&quot;</code></td>
<td>This is a useful one if you want to ensure you’re watching all Patient-related trackers as it will catch those recategorized as Patient-related even after submission.</td>
</tr>
<tr>
<td>All unresolved Patient Care items</td>
<td><code>status in (Triaged, &quot;Waiting for Input&quot;) AND &quot;Work Group&quot; = &quot;Patient Care [pc]&quot;</code></td>
<td>Note this isn’t driven by resolution - it’s driven by status, as preliminary resolutions might be specified on unapproved items</td>
</tr>
<tr>
<td>All unresolved trackers with negative ballot comments</td>
<td><code>status in (Triaged, &quot;Waiting for Input&quot;) AND &quot;Outstanding Negatives&quot; != null</code></td>
<td></td>
</tr>
</tbody>
</table>

Image: [Jira search screenshot]
Unapplied changes pending for more than one month

status = "Resolved - change required" AND updated <= -1m

Technically this just shows items that are unchanged for more than one month - so adding a comment will drop an issue out of this list for another month even though it doesn't change the pending duration.

In addition, there's a custom function we've created specifically to help search on ballots called `inBallot`.

The syntax is as follows: `issueFunction in inBallot("[ballot]", "[filters]")`

The 'ballot' needs to be the ballot code, either just the cycle or the full ballot code. E.g. `2020-Sep - FHIR IG HReX R1 STU` or `2020-Sep - FHIR IG HReX R1 STU`. (It's also possible to just search on 'FHIR IG HReX R1 STU', but that's not so useful.)

The filters is a comma-separated list consisting of either "AFFIRM" or "NEG and/or "OPEN". The first will filter to only included either affirmative or negative votes. The latter will filter to only include balloted issues that have not yet been resolved. (Note - when someone has withdrawn a negative comment, it will no longer be treated as a negative.)

### Examples

<table>
<thead>
<tr>
<th>Description</th>
<th>JQL</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>All comments tied to the Sept. 2020 HReX ballot</td>
<td><code>issueFunction in inBallot(&quot;2020-Sep - FHIR IG HReX R1 STU&quot;)</code></td>
<td></td>
</tr>
<tr>
<td>Only negative votes tied to the Sept. 2020, HReX ballot</td>
<td><code>issueFunction in inBallot(&quot;2020-Sep - FHIR IG HReX R1 STU&quot;, &quot;NEG&quot;)</code></td>
<td>If an issue is withdrawn, it's no longer considered negative</td>
</tr>
<tr>
<td>Unresolved votes tied to the Sept. 2020, HReX ballot</td>
<td><code>issueFunction in inBallot(&quot;2020-Sep - FHIR IG HReX R1 STU&quot;, &quot;OPEN&quot;)</code></td>
<td>Issues are considered 'resolved' if retracted or if a resolution has been voted for them</td>
</tr>
<tr>
<td>Unresolved negative votes tied to the Sept. 2020, HReX ballot</td>
<td><code>issueFunction in inBallot(&quot;2020-Sep - FHIR IG HReX R1 STU&quot;, &quot;NEG,OPEN&quot;)</code></td>
<td>NEG and OPEN can be in either order</td>
</tr>
<tr>
<td>All negatives that were submitted as part of the 2020 January cycle</td>
<td><code>issueFunction in inBallot(&quot;2021-Jan&quot;, &quot;NEG&quot;)</code></td>
<td>Not necessarily super useful to say, but possible</td>
</tr>
</tbody>
</table>

### Favorite searches and other searches

Once you've defined a search, you can choose to save the search so it's available to you in the future. Simply click the Save as button right beside the Search label at the top of the screen. Once saved, you can also click on the Details link which allows giving your search a description, lets you share the search with others and allows you to create a subscription.

Administrators and Managers can also share saved searches, making them available with others. It's possible to see what searches have been shared with you as well as searches you've saved yourself by clicking on the Find filters link right under the New Search button in the left-hand panel on the search screen. If there's a search created by yourself or others that you find particularly useful, you can click on the icon to mark the item as a favorite. Favorites will appear in your search panel at the bottom under "Favorite Filters".

With saved filters, you can also change what columns appear in the list view (including what order the columns are in) by clicking on the Columns button. Be sure to choose "Filter", or the changes will instead be applied to all of your queries as a default (where you haven't overridden column behavior). Also make sure you re-save your search if you've changed the column layout.

### Exporting

Once you've run a query, you have the option of exporting the results. With the button, you can save a summary view or full view of the issues in a wide variety of formats suitable for human review (MS Word, HTML) or programmatic manipulation (XML, JSON).

### Subscriptions

Once a filter has been saved (by you or someone who's shared it with you), you can "subscribe" to it. That means that you can have Jira run the query on a regular basis and email you the results. Note that the filter will run the search each time, so if you only want to see "new" data, you'll need to filter based on updated or some other time-based field. Also be aware that while Jira allows you to specify a specific time for the query to be run and the email to be sent, the actual execution can be slightly variable depending on how busy the server is. That means that if you ask for the query to run daily and to only include issues changed in the last 24 hours, it's possible that you might miss an issue or two if the queries don't run exactly 24 hours apart (so you might want to give yourself a buffer of an extra hour or so, even if that means that your emails may occasionally contain redundant information.)

If you think there's a need for a group of people to receive the same email report, contact an administrator. They have the ability to set up shared subscriptions when appropriate.

### Bulk changes
The last thing you can do from the Search screen is to perform manipulations on the issues returned in response to a search. There’s a maximum limit of 1000 resources, so if your search returns more than that, you’ll need to narrow the search (e.g. by time range) and do the modification in chunks. To perform a bulk operation:

1. select Tools, and beneath Bulk Change click on either the complete count of issues or the current page worth of issues.
2. The next screen will let you select all of the issues or some subset of them (if you wanted to exclude some of the search set from the change). Be sure to wait until the page has fully rendered before attempting to click on the "select all" check-box, as it won't work until all issues have been loaded onto the screen. When done, click Next.
3. This screen will allow you to choose what type of change you want to perform:
   a. Watch or Unwatch are simple options. This is a quick way to watch (or stop watching) a large number of issues
   b. Transition allows you to make a variety of state transitions. Essentially this means duplicating the behavior of hitting one of the "Workflow" buttons on a whole bunch of issues at once. The set of available transitions will depend on what choices are possible for the set of issues included. The ones most appropriate for bulk change for most users are: In Person, Not in Person, Vote Affirmative, Vote Negative, Unvote, Retract or Withdraw. Managers will have additional options.
      i. If you choose one of the vote options, you’ll be prompted to specify the ballot to attach the vote to
4. You’ll then be prompted to confirm your change.
5. In some cases, business rules may prevent all changes from occurring (e.g. if you opted to vote on all of the issues for a particular ballot, but that ballot wasn't relevant for all issues.

Note that while you can cancel a block operation in progress, doing so doesn't rollback changes already made prior to the cancelation.

**WARNING** - DO NOT bulk change the specification for an issue unless you also change the pages, artifacts and versions at the same time. Changing the specification without also updating those other elements will create 'broken' issues and will lead to problems with balloting and other Jira processes.