

Reindex a controlled number of issues in Jira

This can be achieved via the ScriptRunner Console:

```
import com.atlassian.jira.component.ComponentAccessor
import com.atlassian.jira.issue.search.SearchException
import com.atlassian.jira.issue.search.SearchResults
import com.atlassian.jira.issue.search.SearchProvider
import com.atlassian.jira.web.bean.PagerFilter
import com.atlassian.jira.security.JiraAuthenticationContext
import com.atlassian.jira.bc.issue.search.SearchService.ParseResult
import com.atlassian.jira.bc.issue.search.SearchService
import com.atlassian.jira.issue.index.IssueIndexingService
import com.atlassian.jira.util.ImportUtils

JiraAuthenticationContext authenticationContext = ComponentAccessor.getJiraAuthenticationContext()
def authContext = ComponentAccessor.getJiraAuthenticationContext()
def issueIndexingService = ComponentAccessor.getComponent(IssueIndexingService)
def issueManager = ComponentAccessor.getIssueManager()

String jql="project=TEST and status not in (Closed,Resolved)"
Number maxIndexCount = 600

SearchService searchService = ComponentAccessor.getComponentOfType(SearchService.class)
ParseResult parseResult = searchService.parseQuery(authenticationContext.getLoggedInUser(), jql)
int totalIssues = 0

if (parseResult.isValid())
{
    SearchResults results = searchService.search(authenticationContext.getLoggedInUser(), parseResult.
getQuery(), PagerFilter.getUnlimitedFilter())
    final List issues = results?.results

    //Loop all issues
    issues.each { theIssue ->

        if (totalIssues < maxIndexCount)
        {
            boolean wasIndexing = ImportUtils.isIndexIssues()
            ImportUtils.setIndexIssues(true)
            issueIndexingService.reIndex(issueManager.getIssueObject(theIssue.id))
            ImportUtils.setIndexIssues(wasIndexing)
            totalIssues= totalIssues +1

            log.warn "Indexed: " + theIssue.getKey()
        }
    }
}
```