Add Organisations to the Issue for JIRA Service Desk (3.x)

This script adds all Organisations for a User to a Ticket that is being created in Jira service Desk - Created via JIRA, not the Portal. It uses the Reporter.

Notice, a User can belong to several Organisations, all will be added to the Ticket:

Find v4.x of the script here

Thanks to Adaptavist for the Script 🙂

Notice the LimitedPagedRequestImpl(0, 100, 500) lines, if the numbers are to small, nothing will be returned

This is a potential expensive call, if You have a lot or organisations, currently I have a customer with 600 Orgs, and the script takes 3-5 seconds to execute. So we had to change the idea of a scripted field to another solution, as each Issue view took 5-6 seconds.

I assume this will be better in JIRA 8 where the "Atlassian Fugue" is replaced by a proper API: https://confluence.atlassian.com/servicedesk/jira-service-desk-4-0-x-release-notes-961777466.html

Not working for JIRA 8.x

```
import com.atlassian.fugue.Option
import com.atlassian.jira.component.ComponentAccessor
import com.atlassian.jira.issue.ModifiedValue
import com.atlassian.jira.issue.MutableIssue
import com.atlassian.jira.issue.util.DefaultIssueChangeHolder
import com.atlassian.servicedesk.api.ServiceDeskManager
import com.atlassian.servicedesk.api.organization.CustomerOrganization
import com.atlassian.servicedesk.api.organization.OrganizationService
import com.atlassian.servicedesk.api.organization.OrganizationsQuery
import com.atlassian.servicedesk.api.organization.UsersInOrganizationQuery
import com.atlassian.servicedesk.api.util.paging.LimitedPagedRequest
import com.atlassian.servicedesk.api.util.paging.LimitedPagedRequestImpl
import com.onresolve.scriptrunner.runner.customisers.PluginModule
import com.onresolve.scriptrunner.runner.customisers.WithPlugin

@WithPlugin("com.atlassian.servicedesk")

@PluginModule
ServiceDeskManager serviceDeskManager

@PluginModule
OrganizationService organizationService
MutableIssue issue = issue
def currentUser        = ComponentAccessor.getJiraAuthenticationContext().getLoggedInUser()
def serviceDeskProject = serviceDeskManager.getServiceDeskForProject(issue.projectObject)
def serviceDeskId      = serviceDeskProject.right()?.get()?.serviceDeskId as Integer
if (!serviceDeskId) {
    log.debug "Could not find service desk id ${serviceDeskId}, maybe the project ${issue.projectObject.key} is
not a service desk one."
    return
}
def organizationsToSet = []
// get the available organizations for that project
def organizationsQuery = new OrganizationsQuery() {
    @Override
    Option<Integer> serviceDeskId() {
        return new Option.Some<Integer>(serviceDeskId)
    }
    @Override
    LimitedPagedRequest pagedRequest() {
        return new LimitedPagedRequestImpl(0, 100, 500)
    }
}
// iterate over the organizations and if the reporter is part of any keep this organization in a collection
organizationService.getOrganizations(currentUser, organizationsQuery).right().get().results.each { organization
->
    def usersInOrganizationQuery = new UsersInOrganizationQuery() {
        @Override
        CustomerOrganization organization() {
            return organization
        }
        @Override
        LimitedPagedRequest pagedRequest() {
            return new LimitedPagedRequestImpl(0, 100, 500)
        }
    }
    if (issue.reporter in organizationService.getUsersInOrganization(currentUser, usersInOrganizationQuery).
right().get().results) {
        organizationsToSet.add(organization)
    }
}
def cf = ComponentAccessor.getCustomFieldManager().getCustomFieldObjectByName("Organizations")
cf.updateValue(null, issue, new ModifiedValue(issue.getCustomFieldValue(cf), organizationsToSet), new
DefaultIssueChangeHolder())
```