Add Organisations to the Issue for JIRA Service Desk (4.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 v3.x of the script here

```
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.jira.user.util.UserManager
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.atlassian.servicedesk.api.util.paging.PagedRequest
import com.atlassian.servicedesk.internal.feature.organization.api.OrganizationsQueryImpl
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 issue2 = issue

//Use a static Agent, otherwise this will fail for Non-agents if used in Create Post-Function
def userManager        = ComponentAccessor.getUserManager()
def currentUser        = ComponentAccessor.userManager.getUserByKey("Service Desk")
//def currentUser        = ComponentAccessor.getJiraAuthenticationContext().getLoggedInUser()
def serviceDeskProject = serviceDeskManager.getServiceDeskForProject(issue.projectObject)
def serviceDeskId      = serviceDeskProject.getId()
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 = []
def allOrganisations = []

def size = 50
def start = 0
while (size > 0) {
    // get the available organizations for that project
    OrganizationsQuery.Builder builder = new OrganizationsQueryImpl.Builder()
    builder.serviceDeskId(serviceDeskId)
    builder.pagedRequest(new PagedRequest() {
        @Override
        int getStart() {
            return start
        }

        @Override
        int getLimit() {
            return 50
        }
    })
    def organizationsQuery = builder.build()
```

```
    def foundResults = organizationService.getOrganizations(currentUser, organizationsQuery).results
    if (foundResults.size() > 0) {
        allOrganisations.addAll(foundResults)
        start += size
    } else {
        size = 0
    }
}


// iterate over the organizations and if the reporter is part of any keep this organization in a collection
allOrganisations.each { organization ->
    def usersInOrganizationQuery = new UsersInOrganizationQuery() {
        @Override
        CustomerOrganization organization() {
            return organization
        }
        @Override
        LimitedPagedRequest pagedRequest() {
            return new LimitedPagedRequestImpl(0, 10000, 50000)
        }
    }
    if (issue.reporter in organizationService.getUsersInOrganization(currentUser, usersInOrganizationQuery).
results) {
        organizationsToSet.add(organization)
    }

}
if (organizationsToSet.size() >0)
{
        def cf = ComponentAccessor.getCustomFieldManager().getCustomFieldObjectByName("Organizations")
        cf.updateValue(null, issue2, new ModifiedValue(issue.getCustomFieldValue(cf), organizationsToSet), new
DefaultIssueChangeHolder())
}
```