

# Groovy scripts for TTS fields



[See our new groovy scripts document.](#)

In this section, you can find Groovy scripts to get detailed info of TTS custom fields.

If you're using Script Runner Plugin, you can easily use these code snippets to get data.

- [1 Script for Time to SLA Customfield](#)
- [2 Script for SLA Overview Customfield](#)
- [3 Script for Overdue Customfield](#)
- [4 Script for SLA Indicator Customfield](#)

## Script for Time to SLA Customfield

```

import com.atlassian.jira.component.ComponentAccessor
import com.atlassian.jira.issue.CustomFieldManager
import com.atlassian.jira.issue.MutableIssue
import com.atlassian.jira.issue.customfields.CustomFieldType
import com.atlassian.jira.issue.fields.CustomField
import java.util.*
import java.text.SimpleDateFormat
import java.sql.Timestamp

// get CustomFieldManager instance
def customFieldManager = ComponentAccessor.getCustomFieldManager()
// find your TTS custom field's ID and put it here instead of 10600
def ttsField = customFieldManager.getCustomFieldObject("customfield_10600")
// get custom field value object
def ttsFieldValue = issue.getCustomFieldValue(ttsField)
// date/time formatter will be used to format date attributes
def formatter = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
// all attributes will be stored to result variable
def result = ""
// TTS custom field returns list, let's iterate
if (ttsFieldValue && ttsFieldValue.size() > 0) {
    ttsFieldValue.each {
        String slaName = it.slaName
        int slaValueAsMinutes = it.slaValue
        String slaValueAsTimeString = it.slaValueAsString
        String originStatusName = it.originStatusName
        String targetStatusName = it.targetStatusName
        Date originDate = it.statusDate
        Date expectedTargetDate = it.slaTargetDate
        Date actualTargetDate = it.untilDate
        long timeLeftTillSla = it.timeToSla // if less than 0, overdue
        String timeLeftTillSlaAsString = it.timeToSlaAsString // if there is overdue, overdue string
        boolean isPaused = it.paused
        boolean startDateProvidedByDateCustomField = it.startDateProvidedByDateCustomField
        boolean endDateProvidedByDateCustomField = it.endDateProvidedByDateCustomField
        boolean negotiationDateProvidedByDateCustomField = it.negotiationDateProvidedByDateCustomField

        result = ""
        [SLA Name: $slaName]
        [SLA Value As Minutes: $slaValueAsMinutes]
        [SLA Value As Time String: $slaValueAsString]
        [Origin Status: $originStatusName]
        [Target Status: $targetStatusName]
        [Origin Date: ${formatter.format(originDate)}]
        [Expected Target Date: ${expectedTargetDate ? formatter.format(expectedTargetDate) : 'Not yet
defined'}]
        [Actual Target Date: ${actualTargetDate ? formatter.format(actualTargetDate) : 'Not yet'}]
        [Time Left Till SLA as millis: $timeLeftTillSla]
        [Time Left Till SLA as Time String: $timeLeftTillSlaAsString]
        [Is SLA in Paused Status: $isPaused]
        [Is Start Date Provided By Date Custom Field: $startDateProvidedByDateCustomField]
        [Is End Date Provided By Date Custom Field: $endDateProvidedByDateCustomField]
        [Is Negotiation Date Provided By Date Custom Field: $negotiationDateProvidedByDateCustomField]
        <br>
        ""
    }
}
result.toString()

```

### Sample Result

```
[SLA Name: Sla to Resolution]
[SLA Value As Minutes: 240]
[SLA Value As Time String: 4h]
[Origin Status: Open]
[Target Status: Resolved]
[Origin Date: 06/13/2016 09:33:30]
[Expected Target Date: 06/13/2016 13:33:30]
[Actual Target Date: Not yet]
[Time Left Till SLA as milis: -34676000]
[Time Left Till SLA as Time String: - 9 hours, 37 minutes, 56 seconds]
[Is SLA in Paused Status: false]
[Is Start Date Provided By Date Custom Field: false]
[Is End Date Provided By Date Custom Field: false]
[Is Negotiation Date Provided By Date Custom Field: false]
```

### Script for SLA Overview Customfield

```

import com.atlassian.jira.component.ComponentAccessor
import com.atlassian.jira.issue.CustomFieldManager
import com.atlassian.jira.issue.MutableIssue
import com.atlassian.jira.issue.customfields.CustomFieldType
import com.atlassian.jira.issue.fields.CustomField
import java.util.*
import java.text.SimpleDateFormat
import java.sql.Timestamp

// get CustomFieldManager instance
def customFieldManager = ComponentAccessor.getCustomFieldManager()
// find your SLA Overview field's ID and put it here instead of 10801
def overviewField = customFieldManager.getCustomFieldObject("customfield_10801")
def overviewFieldValue = issue.getCustomFieldValue(overviewField)
// date/time formatter will be used to format date attributes
def formatter = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
def returnString = ""
// SLA Overview field returns list
if (overviewFieldValue && overviewFieldValue.size() > 0) {
    // if there is overdue, then all details of overdues are in 'overdues' attribute
    overviewFieldValue.each {
        String slaName = it.sla.description
            def slaStatus = it.status
                int slaValueAsMinutes = it.sla.slaValue
                    String slaValueAsTimeString = it.slaValueAsTimeString
                        Date originDate = it.originDate
                            Date expectedTargetDate = it.expectedTargetDate
                                Date actualTargetDate = it.actualTargetDate
                                    String workingDurationAsTimeString = it.workingDuration
                                        long workingDurationAsSeconds = it.workingDurationAsSeconds
                                            String pausedDurationAsTimeString = it.pausedDuration
                                                // if there is overdue
                                                    String overdueString = it.breachString
                                                        int originStatusId = it.sla.originStatusId
                                                            int targetStatusId = it.sla.targetStatusId
                                                                boolean negotiationDateProvidedByCustomField = it.negotiationDateProvidedByCustomField

returnString += ""
        [SLA Name: $slaName]
            [SLA Status: $slaStatus]
                [SLA Value As Minutes: $slaValueAsMinutes]
                    [SLA Value As Time String: $slaValueAsTimeString]
                        [Origin Date: ${formatter.format(originDate)}]
                            [Expected Target Date: ${expectedTargetDate ? formatter.format(expectedTargetDate) : 'Not yet
defined'}]
                                [Actual Target Date: ${actualTargetDate ? formatter.format(actualTargetDate) : 'Not yet'}]
                                    [Working Duration: $workingDurationAsTimeString]
                                        [Working Duration As Seconds: $workingDurationAsSeconds]
                                            [Paused Duration: $pausedDurationAsTimeString]
                                                [Overdue: $overdueString]
                                                    [Origin Status Id: $originStatusId]
                                                        [Target Status Id: $targetStatusId]
                                                            [Is Negotiation Date Provided By Custom Field: $negotiationDateProvidedByCustomField]
                                                                <br>
returnString += ""
        ]
    }
}

returnString.toString()

```

```
[SLA Name: Sla to Resolution]
[SLA Status: EXCEED]
[SLA Value As Minutes: 240]
[SLA Value As Time String: 4h]
[Origin Date: 06/13/2016 09:33:30]
[Expected Target Date: 06/13/2016 13:33:30]
[Actual Target Date: Not yet]
[Working Duration: 8 hours, 23 minutes, 16 seconds]
[Working Duration As Seconds: 51796]
[Paused Duration: 0]
[Overdue: 4 hours, 23 minutes, 16 seconds]
[Origin Status Id: 49]
[Target Status Id: 53]
[Is Negotiation Date Provided By Custom Field: false]
```

```
[SLA Name: Trivial Sla]
[SLA Status: STILL]
[SLA Value As Minutes: 1440]
[SLA Value As Time String: 1d]
[Origin Date: 06/13/2016 09:33:30]
[Expected Target Date: 06/14/2016 23:54:28]
[Actual Target Date: Not yet]
[Working Duration: 10 minutes]
[Working Duration As Seconds: 600]
[Paused Duration: 1 hour, 13 minutes, 16 seconds]
[Overdue: null]
[Origin Status Id: 49]
[Target Status Id: 54]
[Is Negotiation Date Provided By Custom Field: false]
```

## Script for Overdue Customfield

```

import com.atlassian.jira.component.ComponentAccessor
import com.atlassian.jira.issue.CustomFieldManager
import com.atlassian.jira.issue.MutableIssue
import com.atlassian.jira.issue.customfields.CustomFieldType
import com.atlassian.jira.issue.fields.CustomField
import java.util.*
import java.text.SimpleDateFormat
import java.sql.Timestamp

def customFieldManager = ComponentAccessor.getCustomFieldManager()
// find your Overdue Status custom field's ID and put it here instead of 10601
def overdueField = customFieldManager.getCustomFieldObject("customfield_10601")
def overdueFieldValue = issue.getCustomFieldValue(overdueField)
def formatter = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
def result = ""
// overdue field returns list
if (overdueFieldValue && overdueFieldValue.size() > 0) {
    overdueFieldValue.each {
        String slaName = it.description
        int slaValueAsMinutes = it.slaValue
        String slaValueAsString = it.slaValueAsString
        String originStatusName = it.beforeState
        String targetStatusName = it.afterState
        Date originDate = it.beforeDate
        Date expectedTargetDate = it.expectedTargetDate
        Date actualTargetDate = it.afterDate
        long workingTimeAsMillis = it.workingTimeAsMillis
        String overdue = it.overdue
        boolean startDateProvidedByDateCustomField = it.startDateProvidedByDateCustomField
        boolean endDateProvidedByDateCustomField = it.endDateProvidedByDateCustomField
        boolean negotiationDateProvidedByDateCustomField = it.negotiationDateProvidedByDateCustomField

        result = ""
        [SLA Name: $slaName]
        [SLA Value As Minutes: $slaValueAsMinutes]
        [SLA Value As Time String: $slaValueAsString]
        [Origin Status: $originStatusName]
        [Target Status: $targetStatusName]
        [Origin Date: ${formatter.format(originDate)}]
        [Expected Target Date: ${expectedTargetDate ? formatter.format(expectedTargetDate) : 'Not yet
defined'}]
        [Actual Target Date: ${actualTargetDate ? formatter.format(actualTargetDate) : 'Not yet'}]
        [Overdue: ${overdue}]
        [Is Start Date Provided By Date Custom Field: $startDateProvidedByDateCustomField]
        [Is End Date Provided By Date Custom Field: $endDateProvidedByDateCustomField]
        [Is Negotiation Date Provided By Date Custom Field: $negotiationDateProvidedByDateCustomField]
        <br>
        ""
    }
}
result.toString()

```

## Sample Result

```
[SLA Name: Sla ro Resolution]
[SLA Value As Minutes: 240]
[SLA Value As Time String: 4h]
[Origin Status: Open]
[Target Status: Resolved]
[Origin Date: 06/13/2016 12:27:32]
[Expected Target Date: 06/13/2016 16:27:32]
[Actual Target Date: 06/13/2016 18:28:49]
[Overdue: 2 hours, 1 minute, 17 seconds]
[Is Start Date Provided By Date Custom Field: false]
[Is End Date Provided By Date Custom Field: false]
[Is Negotiation Date Provided By Date Custom Field: false]
```

## Script for SLA Indicator Customfield

```

import com.atlassian.jira.component.ComponentAccessor
import com.atlassian.jira.issue.CustomFieldManager
import com.atlassian.jira.issue.MutableIssue
import com.atlassian.jira.issue.customfields.CustomFieldType
import com.atlassian.jira.issue.fields.CustomField
import java.util.*
import java.text.SimpleDateFormat
import java.sql.Timestamp

def customFieldManager = ComponentAccessor.getCustomFieldManager()
// find your SLA Indicator field's ID and put it here instead of 10700
def indicatorField = customFieldManager.getCustomFieldObject("customfield_10700")
def indicatorFieldValue = issue.getCustomFieldValue(indicatorField)
def formatter = new SimpleDateFormat("MM/dd/yyyy HH:mm:ss")
def returnString = ""
// SLA Indicator field returns an object
if (indicatorFieldValue) {
    String result = indicatorFieldValue.result

    // if there is overdue, then all details of overdues are in 'overdues' attribute
    def overdues = indicatorFieldValue.overdues
    if (overdues.size() > 0) {
        overdues.each {
            def overdue = it.value
            String slaName = overdue.sla.description
            int slaValueAsMinutes = overdue.sla.slaValue
            String slaValueAsTimeString = overdue.sla.slaValueAsString
            Date originDate = overdue.originDate
            Date expectedTargetDate = overdue.expectedTargetDate
            Date actualTargetDate = overdue.actualTargetDate
            String workingDurationAsString = overdue.workingDuration
            long workingDurationAsSeconds = overdue.workingDurationAsSeconds
            String pausedDurationAsString = overdue.pausedDuration
            String overdueString = overdue.breachString
            int originStatusId = overdue.sla.originStatusId
            int targetStatusId = overdue.sla.targetStatusId
            boolean negotiationDateProvidedByCustomField = overdue.negotiationDateProvidedByCustomField

            returnString = """
                [SLA Name: $slaName]
                [SLA Value As Minutes: $slaValueAsMinutes]
                [SLA Value As Time String: $slaValueAsTimeString]
                [Origin Date: ${formatter.format(originDate)}]
                [Expected Target Date: ${expectedTargetDate ? formatter.format(expectedTargetDate) : 'Not yet
defined'}]
                [Actual Target Date: ${actualTargetDate ? formatter.format(actualTargetDate) : 'Not yet'}]
                [Working Duration: $workingDurationAsString]
                [Working Duration As Seconds: $workingDurationAsSeconds]
                [Paused Duration: $pausedDurationAsString]
                [Overdue: $overdueString]
                [Origin Status Id: $originStatusId]
                [Target Status Id: $targetStatusId]
                [Is Negotiation Date Provided By Custom Field: $negotiationDateProvidedByCustomField]
                <br>
            """
        }
    }
}

returnString.toString()

```



## Sample Result

```
[SLA Name: Sla to Resolution]
[SLA Value As Minutes: 240]
[SLA Value As Time String: 4h]
[Origin Date: 06/13/2016 09:33:30]
[Expected Target Date: 06/13/2016 13:33:30]
[Actual Target Date: Not yet]
[Working Duration: 14 hours, 3 minutes, 10 seconds]
[Working Duration As Seconds: 50590]
[Paused Duration: 0]
[Overdue: 10 hours, 3 minutes, 10 seconds]
[Origin Status Id: 49]
[Target Status Id: 53]
[Is Negotiation Date Provided By Custom Field: false]
```