

Common Request

Fluorine+

The **Common Request** document is an XML schema that contains default fields for mapping Request form values from one system to another. The common request format mirrors what one would see of the Request form with related lists expressed in embedded XML form.

A sample **Common Request** looks like this:

```
<common_request>
  <active />
  <activity_due />
  <additional_assignee_list />
  <approval />
  <approval_history />
  <approval_set />
  <approvers />
  <assigned_to />
  <assignment_group />
  <attachments>
    <attachment>
      <data />
      <size_bytes />
      <file_name />
      <sys_id />
      <content_type />
    </attachment>
  </attachments>
  <business_duration />
  <business_service />
  <calendar_duration />
  <calendar_stc />
  <closed_at />
  <closed_by />
  <close_notes />
  <cmdb_ci />
  <comments />
  <comments_and_work_notes />
  <company />
  <contact_type />
  <correlation_display />
  <correlation_id />
  <delivery_address />
  <delivery_plan />
  <delivery_task />
  <description />
  <due_date />
  <escalation />
  <expected_start />
  <follow_up />
  <group_approvals />
  <group_list />
  <impact />
  <knowledge />
  <location />
  <made_sla />
  <number />
  <opened_at />
  <opened_by />
  <order />
  <parent />
  <price />
  <priority />
  <provider />
  <reassignment_count />
```

```

    <recurring_prices>
      <recurring_price>
        <recurring_frequency />
        <recurring_price />
      </recurring_price>
    </recurring_prices>
    <rejection_goto />
    <requested_date />
    <requested_for />
    <requested_items>
      <requested_item>
        <price />
        <stage />
        <quantity />
        <assigned_to />
        <due_date />
        <cat_item />
        <number />
        <sc_catalog />
      </requested_item>
    </requested_items>
    <request_state />
    <short_description />
    <sla_due />
    <special_instructions />
    <stage />
    <state />
    <sys_class_name />
    <sys_created_by />
    <sys_created_on />
    <sys_domain />
    <sys_domain_path />
    <sys_mod_count />
    <sys_tags />
    <sys_updated_by />
    <sys_updated_on />
    <time_worked />
    <upon_approval />
    <upon_reject />
    <urgency />
    <user_input />
    <variables />
    <watch_list />
    <wf_activity />
    <work_end />
    <work_notes />
    <work_notes_list />
    <work_start />
  </common_request>

```

Embedded records

Embedded records are fields in the Common Request document that cannot be mapped by singular values. Instead, they are entirely different records included inside the Common Request document to be created on the subscribing system.

The following are examples of the embedded records in Common Request:

Attachment Embedded Record

The **attachment** field in the Common Request is an embedded record that, when populated, will create a new attachment on the system receiving the Common Request.

Table Map Source Field	Definition
data	Attachment data in an encoded string format

size_bytes	Measurement of how much the attachment data contains
file_name	Name of the attached file
sys_id	Unique record identifier of the attachment
content_type	Attachments content type (i.e jpeg, png, txt, etc.)

Recurring Price Record

The **recurring price** field in the Common Request is an embedded record that, when populated, will create a new recurring price on the system receiving the Common Request.

Table Map Source Field	Definition
recurring_frequency	Rate at which the recurring prices occurs
recurring_price	Price of the recurring prices

Requested Item Record

The **requested item** field in the Common Request is an embedded record that, when populated, will create a new requested item on the system receiving the Common Request.

Table Map Source Field	Definition
price	Price of the requested item
stage	Development of the requested item
quantity	Amount of requested item
assigned_to	User the requested item is assigned to
due_date	Date the requested item must be fulfilled
cat_item	Catalog item reference
number	Record number of the requested item
sc_catalog	Catalog reference

Variables

To access variables in your [table maps](#) scripts, you can use the [variables](#) object of the record. For example, to put all of a requested item's variables into the description field of the outbound table map record:

```
var variables = current.variables.getElements();
var output = "Variables: \r\n ";
for (var i = 0; i<variables.length; i++) {
    var question = variables[i].getQuestion();
    output = output + (question.getLabel() + ":" + question.getValue() + "\r\n") ;
}

answer = current.description + "\r\n\r\n" + output;
```

Similar topics

- [Common Incident](#)
- [Common Change](#)
- [Common Problem](#)
- [Common Request](#)

Contact Perspectium Support



US: 1 888 620 8880

UK: 44 208 068 5953

support@perspectium.com