

HP Service Manager SLM for Request Management

The Service Level Management function is provided for selected modules in Service Manager.

Unfortunately, the functionality for Request Management is not provided as a built-in option. The solution definition includes the both the system enhancements necessary plus the configuration for Request Management. The configuration can be applied to various environments and specific customer requirements.

Enhancements are required to enable the Service Level Management data collection and measurement processes and allow for the configuration of SLM.

OBJECT ENHANCEMENTS

The object record for the Request Management table OCMQ must be modified to add the SLM variables.

To access the object record select **Tailoring > Document Engine > Objects**, and then search for **ocmq**. Add the following variables on the **Variables/Global** Lists tab.

- \$L.sla.id
- \$L.slo.current.status
- \$L.slo.name
- \$L.slo.initial.state
- \$L.slo.final.state
- \$L.slo.expiration.time

DATA DEFINITION

The database definition record for ocmq must include the SLM fields. To access the database definition, select **Tailoring > Database Dictionary**, and then search for ocmq. Add the following SLM related fields:

- sla.alert1 (date/time)
- sla.target (date/time)
- sla.deadline(date/time)
- sla.breach (logical)
- agreement.id (number)
- agreement.ids (array/number)
- next.breach
- number.vj.alerts (alias for number)
- number.vj.slo (alias for number)
- number.vj.sla (alias for number)
- contract.id (number)

PROCESS REVISION

SLM for Request Management

The process `rmq.open` must be modified to obtain the SLA Agreement. To add the `rad` call to the process record, select **Tailoring > Document Engine > Processes**, and then search for **rmq.open**. On the `rad` table, duplicate the `rmq.open` call in the second entry and replace the first entry with the new `sc.get.sla` call. The `sc.get.sla` call

Process Definition

Process Name: `rmq.open`

Save Cursor Position? Run Standard Process when complete?

Run in Window? Window Title: _____

Initial Expressions | Initial Javascript | **RAD** | Final Expressions | Final Javascript | Next Process

RAD Application: `sc.get.sla` Condition: `filename($L.file)='ocmq'`

Parameter Names	Parameter Values
file	<code>\$L.file</code>
number1	<code>contract.id in \$L.file</code>
numbers	<code>agreement.ids in \$L.file</code>

Post RAD Expressions

Expressions evaluated before RAD call

```
if (filename($L.file)='ocmq') then ($file=$L.file) else ($file=$L.file)
```

RAD Application: `rmq.open` Condition: `true`

Parameter Names	Parameter Values
second.file	<code>\$L.object</code>
text	<code>\$L.exit</code>
boolean1	<code>\$L.bg</code>
cond.input	<code>false</code>

Post RAD Expressions

```
if ($L.exit='normal') then ($L.exit='one')
```

COPY THE SLA TAB TO THE OCMQ FORMS

Locate a suitable SLA tab, such as the tab on `CM.change.implementation`. Copy the tab to the appropriate Request Management forms.

UPDATE THE LINK RECORD FOR OCMQ

Source Field	Target File	Target Field	Query
<code>agreement.id</code>	<code>sla</code>	<code>agreement.id</code>	
<code>number.vj.alerts</code>	<code>Alert</code>	<code>unique.key</code>	<code>alert.type='sla' and file.name='ocmq' and active=false</code>
<code>number.vj.slo</code>	<code>sloresponse</code>	<code>foreign.key</code>	<code>foreign.filename='ocmq'</code>
<code>number.vj.sla</code>	<code>slaactive</code>	<code>foreign.key</code>	<code>foreign.filename='ocmq'</code>

UPDATE THE SLO FORMS

Formats `slo.search`; `slocat.searchg`; `slocat`

- Service Area – add `ocmq` to the value list and Request to the display list

UPDATE THE SLA MODULE CONTROL FORM

Formats slamodulecontrol

- Table Name: Add ocmq to the value list for this field.

CONFIGURING SLA FOR REQUEST MANAGEMENT

After the system modifications are made, configuring SLA for Request Management is similar to configuring for other applications.

CREATE SLO(S)

Service Level Management > Service Level Agreements > SLO Catalog

The SLO is created based on the specific requirements. For instance, there may be Response type targets for Acknowledging within a phase of the workflow and in addition, a similar targets for completing the task. These type of targets can be specified by status values in the request. If the targets are varied, it may require SLOs for each phase in each workflow. If the targets are found in multiple phases or multiple workflows, they may be combined through the conditions specified. Additional information can be found in the Help Contents under *Service Level Objective Catalog*.

CREATE THE SLA

Service Level Management > Service Level Agreements > Create New SLA

The SLA contains the SLOs that are applicable and must be specified in the department record, company record or default SLA when editing the control record..

Additional information can be found in the Help Contents under *Service Level Agreement components*.

CONFIGURE THE APPLICATION

Service Level Management > SLA Administration > Configure Application

Following the system enhancements, SLM can be configured for the ocmq table – Request Management. Enter the appropriate fields for ocmq:

- Table name – ocmq
- Record ID Field – number
- SLAID Field – agreement.ids
- Start Time Field – phase.start.date or requested.date
- End Time Field – close.date
- Customer Field – requestor.name
- Technician Field – assigned.to
- Active Condition – open in \$L.file=true
- Response State Field – status

Enter other values based on the SLA requirements.

EDIT THE CONTROL RECORD

Service Level Management > SLA Administration > Edit Control Record

The overall function of Service Level Management is controlled by the *Enable SLA Application* setting. The Default

SLA is set in the control record and is used when the department or company does not specify an SLA. Additional information can be found in the Help Contents under *SLA Control Record*.

How to get the "sloresponse.active.vj.horz" populated with the sla details?

To have the vj working some extra steps need to be performed these are:

In the display engine "rmq.main.display" the below extra lines need to be added to the initializations tab.

```
cleanup($L.sla.id);cleanup($L.slo.current.status);cleanup($L.slo.name);cleanup($L.slo.initial.s  
tate);cleanup($L.slo.final.state);cleanup($L.slo.expiration.time)
```

```
if (not null(agreement.ids in $L.filed)) then  
($L.record.id.field=jscall("sloDisplay.getRecordIdField",  
filename($L.filed));$L.record.id=$L.record.id.field in $L.filed)
```

```
$L.qryFields={"sla.id", "slo.current.status", "slo.name", "slo.initial.state", "slo.final.state",  
"slo.expiration.time"};$L.sloList=jscall("sloDisplay.getListSLOs", filename($L.filed),  
$L.record.id, agreement.ids in $L.filed, $L.qryFields);$L.sla.id=1 in  
$L.sloList;$L.slo.current.status=2 in $L.sloList;$L.slo.name=3 in  
$L.sloList;$L.slo.initial.state=4 in $L.sloList;$L.slo.final.state=5 in  
$L.sloList;$L.slo.expiration.time=6 in $L.sloList
```