

GENESYS

This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Developer's Guide

Customize Journey Dashboard for Pulse

4/30/2025

Contents

- 1 Customize Journey Dashboard for Pulse
 - 1.1 Prerequisites
 - 1.2 Set Pulse Permissions
 - 1.3 Create Journey Pulse Widgets
 - 1.4 Edit Pulse External Templates
 - 1.5 How to modify the JSON Layout

Customize Journey Dashboard for Pulse

Pulse widgets for Context Services are not editable from the Pulse Template Management UI. To customize widgets for Context Services, you must edit and submit the templates through the Pulse REST API.

Prerequisites

Component	Version			
GMS	8.5.106.xx			
	8.5.102.xx or 8.5.105.01			
Pulse	Important Versions 8.5.103 and 8.5.104 are not supported.			
GAX	8.5.102.18			

Set Pulse Permissions

SAX Dashboard Con	figuration	Routing Parameters Reports	Admin	istration Proactive	Engagement	demo	?
Home > Persons > Persons	> demo Prop	erties		Clone	Delete	Move Te	0
General (DBID: 748)	Permiss	sions		Add Access Group	Add Person	Remo	ve
Member Of		Name	\$	Tenant	☆	Create	
Ranks		Dynamic_Supervisor		Environment			•
Options		👂 GAX EZPulse		Environment			
Permissions		pod_superuser		Environment		V	
Dependencies		pod_supervisor		Environment			
Accessible Objects		🔑 pod_user		Environment			
	•	O Departing		Environment		Image: a filler b = 1	•
	Canc	el			Apply	Save	

Open GAX or Genesys Administrator and navigate to **Configuration > Persons**. Select your user and edit his or her permissions to add the **GAX EZPulse** permissions, then **Save**.

Create Journey Pulse Widgets

Add journey sample layouts to Pulse

Context Services Pulse layout samples are available in the following directory:

- <GMS installation directory>/Pulse_8.5.102 for integration with Pulse 8.5.102 versions.
- <GMS installation directory>/Pulse_8.5.103 for integration with Pulse 8.5.105.01.

To deploy these samples in Pulse, you can use the Genesys Pulse Restful Web Service API.

lormal Basi	c Auth Digest Auth	OAuth 1.0	No environment		× m
http://demo	srv:8040/gax/api/wbr	t/layouts/		POST v	🕑 URL params 🕑 Header
Content-Typ	e	appli	ication/json	Manage pre	esets
Header		Valu	e		
form-data	x-www-form-urlen	coded raw	JSON -		
1 { 2 3 4 5 6 7 8 9 10 11 12 13 14	<pre>definition" : { "dd_def" : { "organizat }, "tenant_dbid" "refresh_inter "layout_type" "is_template" "column" : [{ "categ "is_de "avail "k </pre>	ion" : "1" val" : 15, : "ltDATADEH : false, gory" : "ccDI lta_key" : f able_value" value" : [{	POT", IMENSION", true, : {		

Start a Restful client application (Postman in our example). Submit a POST or PUT API query to GAX to publish the layout sample.

- For Pulse 8.5.102 versions:
 - POST /api/wbrt/layouts/ to deploy a new layout.
 - PUT /api/wbrt/layouts/<id> to update an existing layout.
- For Pulse 8.5.105.01:
 - POST /api/wbrt/templates/ to deploy a new layout.
 - PUT /api/wbrt/templates/<guid> to update an existing layout.

Add your widgets

For Pulse 8.5.102 versions

Custom : CS Widge	ts : +	
THE Add a Widget	Add a Widget Choose a Widget Template	Q CS
298:169 ::: :: CS - Context Services by service type	CS - Context Services by Customized demo of the integration between Pulse and Context Services - Context Services' metrics grouped by media type	e lise and ext ped by
	CS - Context Services by Customized demo of the integration between Pulse and Context Services - Loan Context Services - Loan	e Ise and

Once the templates are published in Pulse, it's very easy. Just navigate to your Pulse Dashboard. The new layouts will be available in the creation wizard.

For Pulse 8.5.105.01 version



Once the templates are published in Pulse, it's very easy. Just navigate to your Pulse Dashboard. The new layouts will be available in the creation wizard.

Edit Pulse External Templates

Retrieve your layout for Pulse 8.5.102

← → C demosrv:8040/gax/?#!/view:pulse.dashboa/d/debug	
Apps S Google Genesys Demo Web Apps	- Brookin Francisch
GAX Dashboard Configuration Routing Parameters Reports Administration	06:167 ::: :: CS - Context Services by media type Journeys started ◆
No Data Available.	voice 0 email 0 chat 0
● n/a Journeys ● n/a Journeys ● n/a States co	<u> </u>

Open Pulse in Genesys Administrator extensions and navigate to your Context Services widget. Append /debug at the end of your URL. The widget and layout IDs will show up in the left corner of each widget.

In our example, 297:168 means that the widget ID is 297 and the layout ID is 168.

http://	demosrv:804()/gax/api/wbrt/la	youts/168	GET 🔻 🗹 UI	RL params 🕑 Headers (1)
Conter	nt-Type		application/json	Manage presets	
Heade	۲		Value		(
Send	Preview	Add to col	lection		Reset
ody	Cookies (3)	Headers (7)	STATUS 200 OK TIME 126 ms		
Prett	y Raw	Preview	JSON XML		
1 2 3 4 5 6 7	{ "id": 1 "state" "re "bo "bo "us	168, ': { equested_stati ody_hash_2": : ody_hash_1": : scn": 1735	us": "stACTIVE", 1992487255, 1992487255,		

Get the layout definition of your widget

Start a Restful client application (Postman in our example). Submit a GET API query to GAX to retrieve the layout associated to your widget ID:

GET http://<host>:<port>/gax/api/wbrt/layouts/<layoutID> URL with the header set to Content-Type= application/json See the Genesys Pulse Restful Web Service API for further details.

[+] Expand the layout example

```
{
    "id": 169,
    "state": {
         "requested status": "stACTIVE",
         "body_hash_2": -383534282,
"body_hash_1": -383534282,
"uscn": 1737
    },
"definition": {
    "". "CS
         "name": "CS - Context Services by service type",
         "description": "Customized demo
of the integration between Pulse and Context Services
- Context Services's metrics grouped by service type",
         "refresh interval": 15,
         "tenant_dbid": 1,
"layout_type": "ltDATADEPOT",
         "enable_delta_snapshots": false,
         "default_widget": {
    "id": 0,
              "size": "1x2",
              "label": "CS - Context Services by service type",
              "view": [
                   {
                        "type": "BarView",
```

```
"column_selector": [
                           "service_started"
                       1.
                       "sorting": [
                           {
                                "is asc": false
                           ļ
                       ]
                  }
             ]
         },
         "template_layout_id": 25,
         "template_column": [
             {
                  "id": "service_started",
                  "type": "ctDATADEPOT",
                  "format": "integer"
                  "category": "ccMEASURE",
"label": "started",
                  "vt": "vINT",
                  "dd column": {
dd_cotomin . "
    "group_name": "cs.service",
    "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                      "aggregate_name": "service.started",
"rollup_range": "DAY",
                       "calculation_type": "COUNT",
                       "criteria name": "service type"
                  }
             },
              {
                  "id": "service started perc",
                  "type": "ctDATADEPOT",
                  "format": "percent",
"category": "ccMEASURE",
"label": "% started",
"vt": "vDBL",
                  "dd_column": {
                       "group_name": "cs.service",
"expr": "#a = #dd_stat(#dd, #organization, #group_name,
#aggregate_name, {#criteria_name, #selected}, #rollup_range, #calculation_type);
#b = #dd_stat(#dd, #organization, #group_name, #aggregate_name, {},
\#rollup_range, \#calculation_type); \#a = T(java.lang.Double).parseDouble(\#a); \#b =
"calculation type": "COUNT",
                       "criteria name": "service type"
                  }
             },
              {
                  "id": "service completed",
                  "type": "ctDATADEPOT",
                  "format": "integer"
                  "category": "ccMEASURE",
"label": "completed",
"vt": "vINT",
"dd_column": {
                       "group name": "cs.service",
                       "expr": "#a = #dd stat(#dd, #organization, #group name,
"rollup_range": "DAY",
```

```
"calculation type": "COUNT",
                         "criteria_name": "service_type"
                   }
              },
               {
                   "id": "service completed perc",
                   "type": "ctDATADEPOT",
                   "format": "percent",
                   "category": "ccMEASURE",
"label": "% completed",
                    "vt": "vDBL",
                    "dd column": {
                         "group_name": "cs.service",
                        "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria name, #selected}, #rollup range, #calculation type); #b = #dd stat(#dd,
#organization, #group_name, #aggregate_name, {}, #rollup_range, #calculation_type);
#a = T(java.lang.Double).parseDouble(#a); #b = T(java.lang.Double).parseDouble(#b); (#b == 0)
? 0 : #a*100/#b",
                         "aggregate name": "service.completed",
                        "rollup range": "DAY",
                        "calculation type": "COUNT",
                         "criteria_name": "service_type"
                   }
              },
               {
                   "id": "service associated",
                   "type": "ctDATADEPOT",
                   "format": "integer"
                    "category": "ccMEASURE",
                   "label": "associated",
                   "vt": "vINT",
"dd_column": {
"group_name": "cs.service",
        "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                        "aggregate_name": "service.associated",
"rollup_range": "DAY",
                         "calculation_type": "COUNT",
                         "criteria name": "service type"
                   }
              },
                   "id": "service durationAVG",
                   "type": "ctDATADEPOT",
                   "format": "time",
                   "category": "ccMEASURE",
"label": "Avg Duration",
                   "vt": "vINT",
                    "dd_column": {
dd_column . {
    "group_name": "cs.service",
    "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                         "aggregate name": "service.durationAVG",
                        "rollup_range": "DAY"
                        "calculation_type": "AVG",
"criteria_name": "service_type"
                   }
              },
                   "id": "service_durationSUM",
                    "type": "ctDATADEPOT",
                   "format": "time",
                   "category": "ccMEASURE",
```

```
"label": "Duration",
                    "vt": "vINT",
                    "dd column": {
                          "group name": "cs.service",
"expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                          "aggregate_name": "service.durationSUM",
                         "rollup range": "DAY"
                         "calculation type": "SUM",
                         "criteria_name": "service_type"
                    }
               },
{
                    "id": "month service started",
                    "type": "ctDATADEPOT",
                    "format": "integer"
                    "category": "ccMEASURE",
"label": "started (30 days)",
                    "vt": "vINT",
                    "dd column": {
                         "group_name": "cs.service",
"expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria name, #selected}, #rollup range, #calculation type)",
                         "aggregate_name": "service.started",
"rollup_range": "MONTH",
                         "calculation_type": "COUNT",
                          "criteria_name": "service_type"
                    }
               },
                    "id": "month service completed",
                    "type": "ctDATADEPOT",
                    "format": "integer",
                    "category": "ccMEASURE",
"label": "completed (30 days)",
"vt": "vINT",
                    "dd_column": {
"group_name": "cs.service",
            "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                         "aggregate_name": "service.completed",
"rollup_range": "MONTH",
                         "calculation_type": "COUNT",
                          "criteria_name": "service_type"
                    }
               },
{
                    "id": "month service completed perc",
                    "type": "ctDATADEPOT",
                    "format": "percent"
                    "category": "ccMEASURE",
"label": "% completed (30 days)",
                    "vt": "vDBL",
                    "dd_column": {
"group_name": "cs.service",
            "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type); #b = #dd_stat(#dd,
#organization, #group_name, #aggregate_name, {}, #rollup_range, #calculation_type); #a =
T(java.lang.Double).parseDouble(#a); #b = T(java.lang.Double).parseDouble(#b); (#b == 0) ? 0
: #a*100/#b",
                          "aggregate name": "service.completed",
                          "rollup range": "MONTH"
                         "calculation_type": "COUNT",
```

```
"criteria_name": "service_type"
         }
    }
],
"column": [
    {
         "id": "_Object$ID",
         "category": "ccDIMENSION",
"selected_value": {
             "type": "stKEYVAL",
             "kvalue": [
                 {
                      "k": "4480",
                     "v": "Auto Care Application"
                 },
                  {
                      "k": "4487",
                      "v": "Auto Care Cancelation"
                 },
                  {
                      "k": "4481",
                      "v": "Auto Care Selection"
                 },
{
                      "k": "4457",
                      "v": "Branch Appointment"
                 },
                  {
                      "k": "4479",
                      "v": "Credit Card Application"
                 },
{
                     "k": "4485",
                      "v": "Credit Card Cancelation"
                 },
                  {
                     "k": "4478",
                      "v": "Credit Card Selection"
                 },
                  {
                     "k": "4458",
                     "v": "General Inquiries"
                 },
                  {
                      "k": "4486",
                      "v": "Home Care Cancelation"
                 },
                  {
                      "k": "4482",
                      "v": "Home Care Selection"
                 },
{
                     "k": "4459",
                      "v": "Loan Application"
                 },
                  {
                      "k": "4484",
                     "v": "Loan Cancelation"
                 },
{
                     "k": "4460",
                      "v": "Loan Selection"
                 }
```

```
]
    },
    "is delta key": true,
    "available_value": {
        "type": "stKEYVAL",
"kvalue": [
            {
                 "k": "4460",
                 "v": "Loan Selection"
            },
             {
                 "k": "4457",
                 "v": "Branch Appointment"
            },
             {
                 "k": "4478",
                 "v": "Credit Card Selection"
            },
{
                 "k": "4479",
                 "v": "Credit Card Application"
             },
             {
                 "k": "4459",
                 "v": "Loan Application"
            },
             {
                 "k": "4485",
                 "v": "Credit Card Cancelation"
            },
             {
                 "k": "4486",
                 "v": "Home Care Cancelation"
            },
{
                 "k": "4487",
                 "v": "Auto Care Cancelation"
             },
             {
                 "k": "4481",
                 "v": "Auto Care Selection"
            },
             {
                 "k": "4480"
                 "v": "Auto Care Application"
            },
{
                 "k": "4482",
                 "v": "Home Care Selection"
            },
             {
                 "k": "4484",
                 "v": "Loan Cancelation"
             },
             {
                 "k": "4458",
                 "v": "General Inquiries"
            }
        ]
    }
},
{
    "id": "_Object$Name",
```

```
"format": "string",
                     "category": "ccDIMENSION"
               },
{
                     "id": "service started",
                     "type": "ctDATADEPOT",
                     "format": "integer"
                     "category": "ccMEASURE",
"label": "started",
                     "vt": "vINT",
                     "dd_column": {
                          "group_name": "cs.service",
"expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                          "aggregate name": "service.started",
                          "rollup_range": "DAY",
"calculation_type": "COUNT",
                          "criteria_name": "service_type"
                     }
               },
{
                     "id": "service_started_perc",
                     "type": "ctDATADEPOT",
                     "format": "percent"
                     "category": "ccMEASURE",
"label": "% started",
                     "vt": "vDBL",
                     "dd column": {
                           'group_name": "cs.service",
                          "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type); #b = #dd_stat(#dd,
#organization, #group_name, #aggregate_name, {}, #rollup_range, #calculation_type); #a =
T(java.lang.Double).parseDouble(#a); #b = T(java.lang.Double).parseDouble(#b); (#b == 0) ? 0
: #a*100/#b",
                          "aggregate name": "service.started",
                          "rollup_range": "DAY",
"calculation_type": "COUNT",
                          "criteria_name": "service_type"
                     }
               },
                     "id": "service completed",
                     "type": "ctDATADEPOT",
                     "format": "integer"
                     "category": "ccMEASURE",
"label": "completed",
"vt": "vINT",
                     "dd column": {
dd_cotdumn . {
    "group_name": "cs.service",
    "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                          "aggregate_name": "service.completed",
"rollup_range": "DAY",
                          "calculation_type": "COUNT",
                          "criteria_name": "service_type"
                     }
               },
{
                     "id": "service completed perc",
                     "type": "ctDATADEPOT",
                     "format": "percent"
                     "category": "ccMEASURE",
"label": "% completed",
```

```
"vt": "vDBL",
                   "dd_column": {
                        "group_name": "cs.service",
"expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type); #b = #dd_stat(#dd,
#organization, #group_name, #aggregate_name, {}, #rollup_range, #calculation_type); #a =
T(java.lang.Double).parseDouble(#a); #b = T(java.lang.Double).parseDouble(#b); (#b == 0) ? 0
: #a*100/#b",
                        "aggregate name": "service.completed",
                        "rollup_range": "DAY"
                        "calculation_type": "COUNT",
"criteria_name": "service_type"
                   }
              },
              {
                   "id": "service associated",
                   "type": "ctDATADEPOT",
                   "format": "integer"
                   "category": "ccMEASURE",
"label": "associated",
                   "vt": "vINT",
                   "dd_column": {
                         "group name": "cs.service",
                        "expr": "#a = #dd stat(#dd, #organization, #group name, #aggregate name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                        "aggregate name": "service.associated",
                        "rollup_range": "DAY"
                        "calculation_type": "COUNT",
"criteria_name": "service_type"
                   }
              },
{
                   "id": "service_durationAVG",
                   "type": "ctDATADEPOT",
                   "format": "time"
                   "category": "ccMEASURE",
"label": "Avg Duration",
                   "vt": "vINT",
                    "dd column": {
                         'group name": "cs.service",
                        "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                        "aggregate_name": "service.durationAVG",
                        "rollup_range": "DAY"
                        "calculation_type": "AVG",
                        "criteria name": "service type"
                   }
              },
               {
                   "id": "service durationSUM",
                   "type": "ctDATADEPOT",
                   "format": "time",
                   "category": "ccMEASURE",
"label": "Duration",
                    "vt": "vINT",
                   "dd column": {
                         "group_name": "cs.service",
                        "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                        "aggregate_name": "service.durationSUM",
"rollup_range": "DAY",
                        "calculation type": "SUM",
                        "criteria_name": "service_type"
```

```
}
               },
                    "id": "month_service_started",
                    "type": "ctDATADEPOT",
                    "format": "integer"
                    "category": "ccMEASURE",
"label": "started (30 days)",
                    "vt": "vINT",
                    "dd column": {
dd_cotdmm1. {
    "group_name": "cs.service",
    "expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                         "aggregate_name": "service.started",
"rollup_range": "MONTH",
                         "calculation_type": "COUNT",
                         "criteria_name": "service_type"
                    }
               },
                    "id": "month_service_completed",
                    "type": "ctDATADEPOT",
                    "format": "integer"
                    "category": "ccMEASURE",
"label": "completed (30 days)",
                    "vt": "vINT",
                    "dd_column": {
                         "group_name": "cs.service",
"expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type)",
                         "aggregate_name": "service.completed",
"rollup_range": "MONTH",
                         "calculation_type": "COUNT",
                         "criteria_name": "service_type"
                    }
               },
{
                    "id": "month_service_completed_perc",
                    "type": "ctDATADEPOT",
"format": "percent",
"category": "ccMEASURE",
"label": "% completed (30 days)",
                    "vt": "vDBL",
                    "dd_column": {
                         "group_name": "cs.service",
"expr": "#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name,
{#criteria_name, #selected}, #rollup_range, #calculation_type); #b = #dd_stat(#dd,
#organization, #group name, #aggregate name, {}, #rollup range, #calculation type); #a =
T(java.lang.Double).parseDouble(#a); #b = T(java.lang.Double).parseDouble(#b); (#b == 0) ? 0
: #a*100/#b",
                         "aggregate name": "service.completed",
                         "rollup range": "MONTH"
                         "calculation_type": "COUNT",
                         "criteria_name": "service_type"
                    }
               }
          "dd def": {
               "organization": "1"
          }
     },
      'generator info": {
          "name": "Pulse",
```

```
"version": "8.5.101.03",
"timestamp": 1467298003720,
"pulse_app_id": 1,
"cme_app_dbid": 366,
"cme_app_name": "GAX_Server"
}
```

Retrieve your layout for Pulse 8.5.105.01

Pulse	×				
	calhost:8040/g	Jax/plugins/pulse/#/	/debug		
Se Pulse GAX					
📰 CS Widgets	÷.				
			:	\backslash	
Context Services					
total					l
Number of services	started		0		
Number of services	completed		0		1
Number of services	associated		0	/	
Average service dura	ation		-1	/	
Total duration of all	services				/
<u> </u>	$\left(\right)$	Widget:1dec84f8.Lavo	put:1dd0bf97	$\mathbf{)}$	
					l
					- A



Open Pulse in Genesys Administrator extensions and navigate to your Context Services widget. Append /debug at the end of your URL and press **Enter**.

- The URL is restored to its previous state (without /debug).
- The widget and layout GUIDs will show up as a clickable link available in the right bottom corner of each widget.

Click the link to display the layout in a separate tab.

Edit and publish the layout

To learn how you can modify the layout, read the detailed section below. Then, deploy or update your layout.

How to modify the JSON Layout

Each Pulse template has a unique ID, a definition, and a state property. If you wish to modify the template, do not modify the template ID; else, if you modify this ID, you will create a new template.

```
{
    "id": <id>,
    "state" : { /** specific state information - do not modify **/
    "definition": {
        /*... See Definition section */
     },
    "generator_info": { /** specific generated information - do not modify **/
    }
}
```

Important

In Pulse version 8.5.105.01, the templates use the guid field instead of id.

Definition property

In the Definition parameter of the JSON layout, you will find the following parameters.

Important

Not all Pulse parameters are listed below.

Field	Description
name	Defines the template name displayed to the User in the Add a Widget interface of Pulse.
	Defines the template description displayed to the User in the Add a Widget interface of Pulse.
description	"Customized demo of the integration between Pulse and Context Services - Context Services's metrics grouped by service type"
layout_type	Defines the source of the layout type. Do not modify this type; it must be set to ltDATADEPOT. DataDepot is an internal collector that Context Services uses to communicate with Pulse.
refresh_interval	Defines the refresh interval of the template in seconds. For instance, 60.
columns	Defines the content of the objects and statistics available in the template. See the columns property table below for further details.

Columns property

The following table details the JSON layout columns properties of the definition attribute.

columns				
Field	Description			
id	Defines the technical alias of the objects or statistics IDs defined in the template_columns property. For instance, "id": "_Object\$ID" or "id": "service_started"			
category	Category of the column that can be a dimension if set to ccDIMENSION, or a statistic if set to ccMEASURE.			
format	Format of this column that can be one of the following values: string, percent, integer, time			
available_value	 Objects available in the template. This field is mandatory for the Edit wizard of the widget when the user selects the object. The kvalue array contains key-value pairs of Business Attribute values to use in this category. 			

Field	Description
	 The k field contains the dbid of the business attributes. The v field contains the label or display name to display for this Business attribute. "available_value": { "type": "stKEYVAL", "kvalue": [{ "v": "Loan Selection",
selected_value	<pre>Objects selected in the template. • This field is mandatory for the Edit wizard of the widget when the user selects the object. • The kvalue array contains key-value pairs of Business Attribute values to use in this category. • The k field contains the dbid of the business attributes. • The v field contains the label or display name to display for this Business attribute. "selected_value": { "type": "stKEYVAL", "kvalue": [{ "v": "Loan Selection",</pre>

Template_column property

For Pulse 8.5.102 versions



The template_column property contains an array of column templates. Each column template defines a statistic to calculate and will be displayed in the **Statistics** panel of the widget creation wizard.

{ /** Other statistics **/
 }

For	Pulse	8.5	105.	01	version
-----	-------	-----	------	----	---------

],

) Pulse	× \				
→ C	 localhost:8040, 	/gax/plugins/pulse/#/			☆ 😵 🚱 🖬
Pulse G	AX				🌣 default 🗸
CS Widgets	s : 🌣 A	dd a Widget			
dd a Wio	dget (CS Widge	ets) > Context Servi	ces Template		
	Objects	Statistics	Display Options		Widget Summery
					widget Summary
Select S	tatistic(s)				Objects (1)
N	lumber of servic	 Display Name			× total
M N	lumber of servic	Number of service	s associated		
	lumber of servic	Alias	Display Format		
		service_associated	d Integer		
	werage service	Aggregate Name	Calculation Type		
Т	otal duration of	Crown Name	Dollyn Dongo		
M N	lumber of states	cs.service	DAY	•	Clear all
N	lumber of states				Statistics (9)
M N	lumber of tasks				× Number of services started
	lumber of tasks				× Number of services completed
					× Average service duration
					× Total duration of all services
					× Number of states started
					× Number of tasks started
					× Number of tasks completed
					Clear all

The template_column property contains an array of column templates. For each dd_column, Pulse displays details on how this column is calculated: type of calculation, roll-up period, aggregate, group names, and more.

Create dd_column objects

Each dd_column object is a JSON Object that defines the statistics data and calculations. The

statistics is calculated within the **expr** mandatory property that is a Spring expression. See the official documentation here.

This expression calls a custom extension function dd_stat to retrieve Statistics and assigns the result to a variable named "a". The expression uses the other properties defined in the dd_column structure to calculate the statistics.

Property Name	Possible Values	Definition
expr	Spring Expression Language (SpEL).	Defines a statistic using the above values to create a Spring expression. The expression uses the properties defined in the dd_column structure: • group_name • aggregate_name • criteria_name • calculation_type
group_name	cs.service	Group to use for Context Services.
aggregate_name	 service.started service.completed service.associated state.started state.completed task.started task.completed service.duration<sum,avg,min< li=""> </sum,avg,min<>	Events used for the statistics.
criteria_name	media_typeservice_typestate_typetask_type	Not mandatory.
calculation_type	COUNT	Counts events.
	SUM MIN MAX AVG	Used for duration statistics (service.duration <sum,avg,min,ma< td=""></sum,avg,min,ma<>
rollup_range	• "QUARTER_HOUR"	Timeframe for the statistic.

Property Name	Possible Values	Definition
	 "HALF_HOUR" 	
	• "HOUR"	
	• "DAY"	
	• "WEEK"	
	• "MONTH"	
	 "QUARTER_YEAR" 	
	• "HALF_YEAR"	
	• "YEAR"	

The dd_column example above retrieves a simple value, but it could report a percentage value. For example, you can report a percentage of services started for one media type compared to all of the started services or media types.

```
#a = #dd_stat(#dd, #organization, #group_name, #aggregate_name, {#criteria_name, #selected},
#rollup_range, #calculation_type);
#b = #dd_stat(#dd, #organization, #group_name, #aggregate_name, {}, #rollup_range,
#calculation_type);
#a = T(java.lang.Double).parseDouble(#a);
#b = T(java.lang.Double).parseDouble(#b);
(#b == 0) ? 0 : #a*100/#b
```

The code above queries two Data Depot metrics, converts each value from String to double, then calculates the required percentage value. As you may have noticed, both expressions used the #selected variable.

Metrics across multiple groups

The Pulse layout definition allows you to query for the same metrics across multiple groups (for example, media types). To do so, you can define a list of all possible values in the structure called available_value. During the widget configuration, the Pulse user will select the available values to display on the dashboard.

The values that the user selects are saved as a subset of the available values in the selected_value structure and later iterated to provide the result.

```
"available_value" : {
    "kvalue" : [{
        "v" : "media type 1",
        "k" : "1"
    }, {
        "v" : "media type 2",
        "k" : "2"
    }, {
        "v" : "media type 3",
        "k" : "3"
    }],
    "type" : "stKEYVAL"
},
"selected_value" : {
```

```
"kvalue" : [{
    "v" : "media type 1",
    "k" : "1"
}, {
    "v" : "media type 2",
    "k" : "2"
}],
"type" : "stKEYVAL"
}
```