Request Fulfillment

General Description

Request Fulfillment is responsible for handling all Service Requests. The term Service Request is used as a general description for different types of requests which users submit to GS & IT.

Request Fulfillment covers the entire scope of possible requests and demands submitted by users, the only exception being Incidents which are handled by Incident Management. The goal of Request Fulfillment is to provide quick and effective access to standard services in order to enable users to improve their productivity by using a standardized, repeatable and formalized process.

NOTE: We are going to redesign Request Fulfillment.

Technical overview

Request Fulfillment is a child of the table *task*, called *u_request_fulfillment*. so it inherits most of the fields we need for request.

It is not state-driven such as incident, because we use the Graphical Workflow engine that automates multi-step processes. The workflow takes into account the different request types based on the category, managing the approvals and transitions based on conditions.

Similarities/Diferences with incident process

We have tried to develop request similar to incident, from the point of view of development these are the similarities:

- Similar form with the common fields such as Functional Element, Assignment group, etc.
- Mechanism for Assign to a person, Rota, etc.
- Flags when there is a new comment from the user, notifications, confidential tickets.
- Metrics
- Even if we have another field for the state, we use the same patterns.

Main differences:

- Created specific state for request similar to incident (field u_current_task_state)
- Depending on a category, we follow a path of execution that can contain different approvals, only go to the Service Desk, etc.
- We don't use SLA together with Service Offerings.
- The urgency is not managed by the CI relations, having as values: As soon as possible, Flexible and Fixed.
- The impact have as values: Essential, Important, Necessary, Useful and Nice to have
- We don't use the priority field.
- Different close codes.

Categories

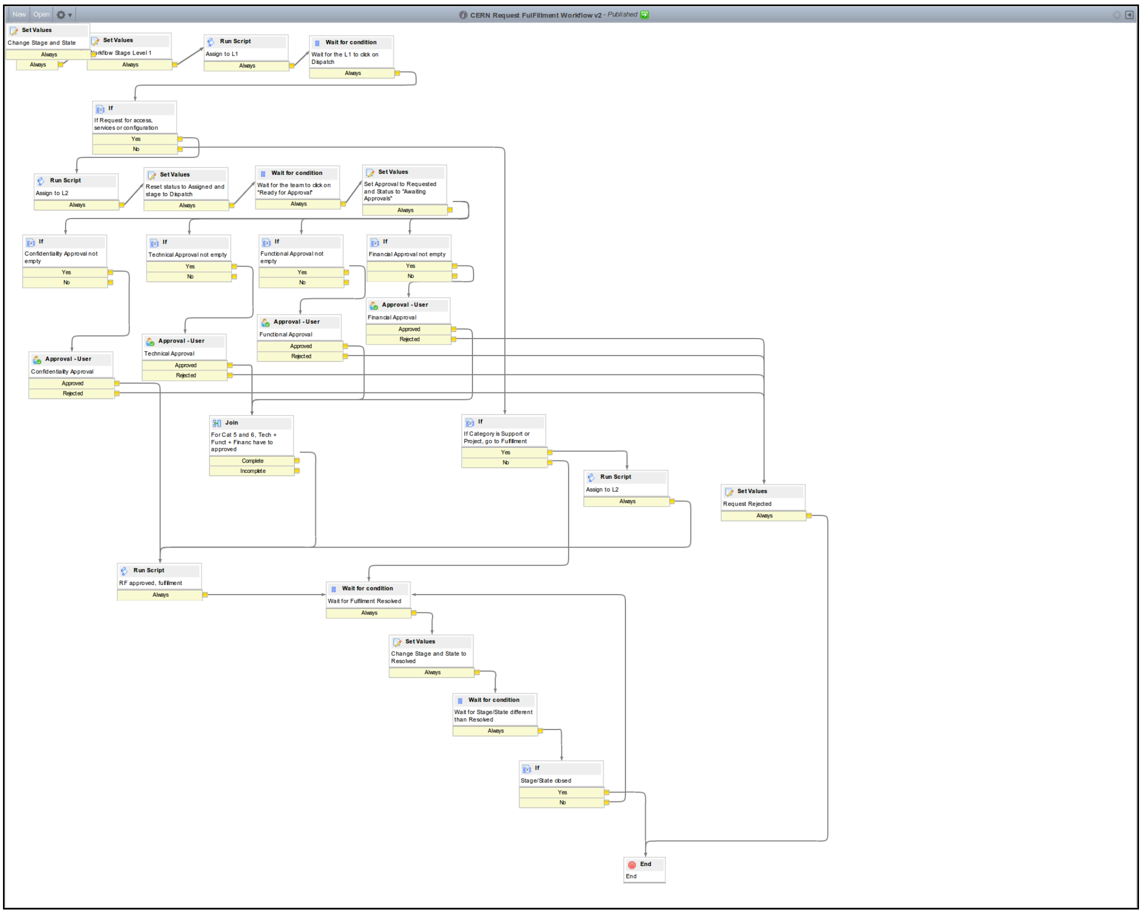
The following categories with a brief overview.

Information	It is only available to be handled by the Service Desk
Support & Consultancy	No approvals, similar to incident
Project	No approvals, similar to incident

Access	"Confidentiality" approval
Services & Products	"Technical", "Functional" & "Financial" approval
Configuration & Enhancements	"Technical", "Functional" & "Financial" approval

Workflow

Current workflow on 5th May 2011, called 'CERN Request Fulfillment Workflow v2':



Coordination of workflow stage, task state and request state

There are three different states that we need to consider on Request fulfillment:

Task state	state	The general state for all tasks.
Request state	u_current_task_state	The specific request state, similar to incident
Workflow stage	u_workflow_stage	Stages are used to provide summary-level feedback about the progress/state of a workflow

Mapping between task state and request state

Located in the business rule 'req_item_state_map':

Request state value	Request state label	Task state label
1	New	Open
2	Assigned	Open
3	Accepted	Open
4	In progress	Work in progress
5	Waiting for user	Pending

Categories

6	Waiting for 3rd party	Pending
7	Waiting for Project Mgmt	Pending
8	Waiting for Change Mgmt	Pending
9	Resolved	Pending
10	Closed	Closed complete (task.active = false)
11	Archived	Closed complete (task.active = false)

Correlation with workflow stage

The stages doesn't have a direct mapping to any state, as it is a more high-level summary about the progress of the workflow. The field is u_workflow_stage of type workflow.

Context

When we started Request Fulfillment, we choose the sc_req_item table. This was a bad decision, as it was a special table for the Request Items concept of Service Now.

This decision brought some problems, as this table was a bit special with the workflows. In this case, the sc_req_item.stage field is defined as a 'workflow' type field. So, when you select this field as the Stage field (highly recommended, btw) the workflow widget will be displayed when the sc_req_item is shown on a list (just as it is now) and it will show the Stages along with their state (Pending, Active, Completed, etc.). More information in here.

There was a bug showing the *nice* view on the list form with the above solution, so we used another field 'u_workflow_stage'. Then, we manually set the stage on some steps on the workflow, being poorly managed and not using the stage mechanism that comes out of the box.

Whe we migrated from sc_req_item to u_request_fulfillment, we didn't change this fact because apparently the bug to show the workflow in the list was only possible with the solution we presented.

It can be researched, if using the standard stage field for the different modules from the workflow and removing the workflow stages feature from the list, it works. It will be handled more easily and maintenable the different stages.

NOTE: I cannot understand how the process flow (the top bar of incident) is generated. The labels are 'Level 1', 'Waiting for approvals', etc. when no direct relation to the values/labels of the workflow stage, default stages of the workflow, etc. It is a detail, but everywhere we have the stage 'Approvals' as value/label, but in the process flow it appears 'Waiting for approvals'.

Values of workflow stages

This values are the choice options of the fields. In the workflow itself there are some Default stages (by table) and also stages for the specific workflow, but we are not using them (read the Context section above and the service now documentation).

Level1	
Dispatch	
Approvals	
Fulfillment	
Resolved	
Closed	

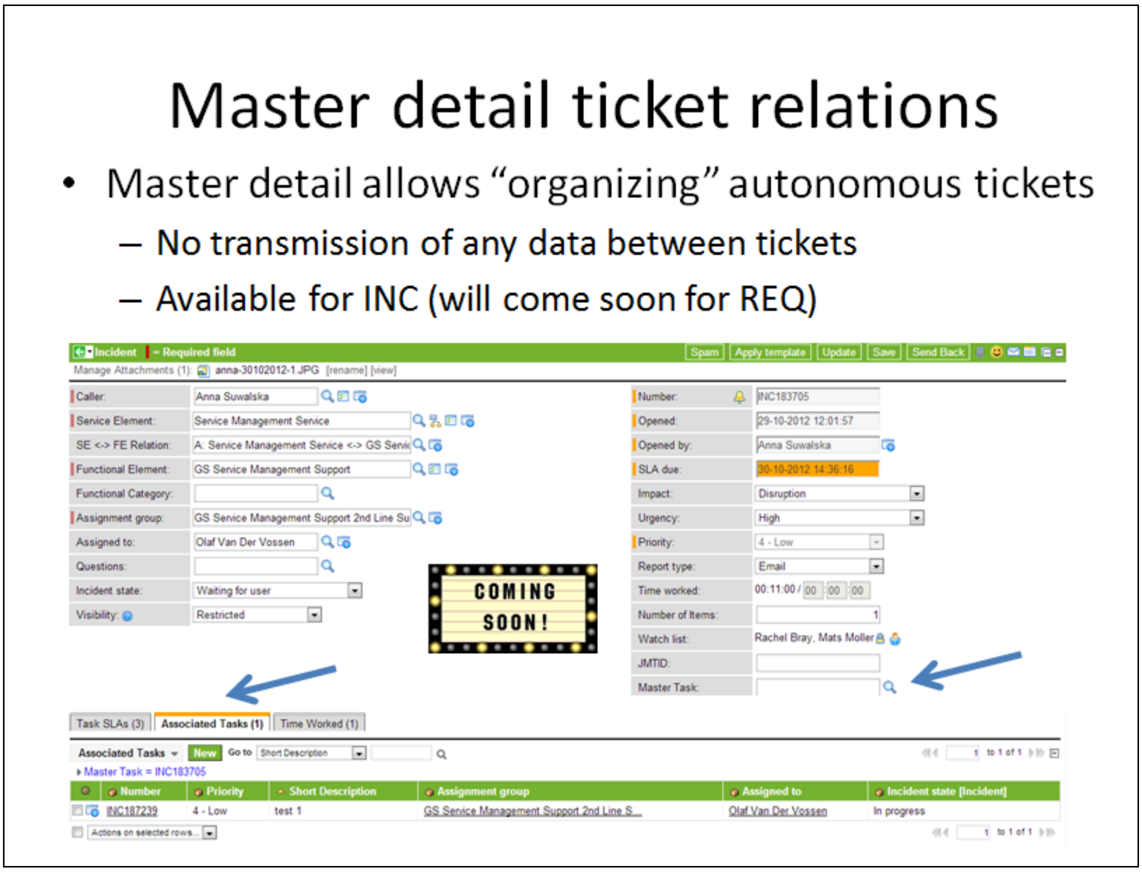
Master-Detail relationship

Master- Detail is only a way to organize tickets, if you receive a request to do multiple things you can create detailed requests and attach them to the master.

Example you can think of: the king of Spain is coming to CERN. You will need (and create separate tickets for): flags, red carpet, guards, medical service, VIP service etc

Note that there is no communication between tickets other than the numbers linking the tickets. They can be for different SE, FE etc..

Note: The parent/child relationship does not exist for Request. The reason is because An incident is something that is broken, multiple people can report it so parent/child helps to solve 1 unique issue for multiple people. A Request is something you want to have or done: If multiple people ask for example for a password reset then there are multiple actions to do for the individuals each having a new password. Yes there could be exceptional cases where two persons ask for the same thing at the same time but we did not encounter that.



Master detail ticket relations

- Master detail allows "organizing" autonomous tickets
  - No transmission of any data between tickets
  - Available for INC (will come soon for REQ)

Incident | = Required field

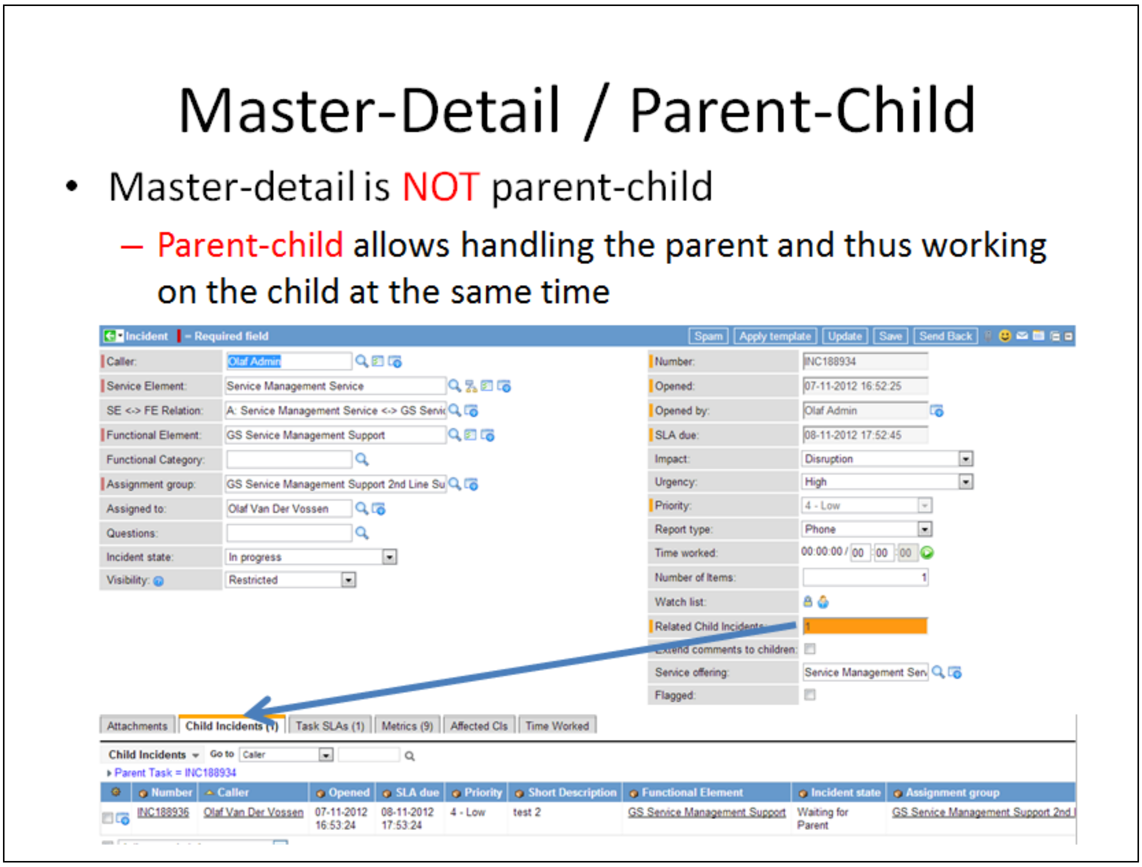
Manage Attachments (1): anna-30102012-1.JPG [rename] [view]

Caller:	Anna Suwalska	Number:	INC183705
Service Element:	Service Management Service	Opened:	29-10-2012 12:01:57
SE <-> FE Relation:	A: Service Management Service <-> GS Servic	Opened by:	Anna Suwalska
Functional Element:	GS Service Management Support	SLA due:	30-10-2012 14:36:16
Functional Category:		Impact:	Disruption
Assignment group:	GS Service Management Support 2nd Line Su	Urgency:	High
Assigned to:	Olaf Van Der Vossen	Priority:	4 - Low
Questions:		Report type:	Email
Incident state:	Waiting for user	Time worked:	00:11:00 / 00 : 00 : 00
Visibility:	Restricted	Number of Items:	1
		Watch list:	Rachel Bray, Mats Moller
		JMTID:	
		Master Task:	

Associated Tasks Go to Short Description

Master Task = INC183705

	Number	Priority	Short Description	Assignment group	Assigned to	Incident state [Incident]
□	INC187239	4 - Low	test 1	GS Service Management Support 2nd Line S...	Olaf Van Der Vossen	In progress



Master-Detail / Parent-Child

- Master-detail is NOT parent-child
  - Parent-child allows handling the parent and thus working on the child at the same time

Caller:	Olaf Admin
Service Element:	Service Management Service
SE <-> FE Relation:	A: Service Management Service <-> GS Servic
Functional Element:	GS Service Management Support
Functional Category:	
Assignment group:	GS Service Management Support 2nd Line Su
Assigned to:	Olaf Van Der Vossen
Questions:	
Incident state:	In progress
Visibility:	Restricted

Number:	INC188934
Opened:	07-11-2012 16:52:25
Opened by:	Olaf Admin
SLA due:	08-11-2012 17:52:45
Impact:	Disruption
Urgency:	High
Priority:	4 - Low
Report type:	Phone
Time worked:	00:00:00 / 00 : 00 : 00
Number of Items:	1
Watch list:	
Related Child Incidents:	1
Extend comments to children:	☐
Service offering:	Service Management Serv
Flagged:	☐

Parent Task = INC188934

	Number	Caller	Opened	SLA due	Priority	Short Description	Functional Element	Incident state	Assignment group
☐	INC188936	Olaf Van Der Vossen	07-11-2012 16:53:24	08-11-2012 17:53:24	4 - Low	test 2	GS Service Management Support	Waiting for Parent	GS Service Management Support 2nd l

Limitations/Problems

- [Important] There was a problem rolling back from Resolved to In progress, we have solved this issue and it has a detailed explanation here. Note that we have started to versioning workflows to cover old requests without the new workflow.
- We combine the Workflow with Business rules and Client Scripts depending on the request state and category, being difficult to know what has changed what.
- We cannot handle exceptions really easily because the workflow has a fixed execution.
- There is step in the Workflow where we need to 'Escalate for Fulfillment' that assigns the ticket to the 2nd Line Support of the current Functional Element. Then you **cannot change the category** because we need to choose a path in the workflow execution (with approvals or not).

Useful information

More information about the processes can be found in the Service Manager's Sharepoint site

-- OmarPera - 05-May-2011

This topic: IT > SNowRequestFulfillment
Topic revision: r9 - 01 Aug 2014 - RamonGarciaVelaCenal

Ideas, requests, problems regarding TWiki? Send feedback