Agiloft ITIL Guide

CONTENTS

ITIL Knowledgebase Guide

This guide describes our out-of-the-box ITIL system.

The ITIL Knowledgebase provides all the functionality to manage a complex IT organization, including Service Request Management, Incident Management, Problem Management, Change Management, Configuration Management, Purchase Management, Project Management, and more.

Our goal has been to provide ready-to-go, out-of-the-box ITIL-compliant structures and process flows for managing IT services, while enabling complete extensibility to meet the needs of any particular organization.

ITIL is a set of best practices intended to improve IT service while reducing failures and costs. The actual implementation of any ITIL process is open to a wide range of interpretations. While the basic structure of Service, Incident, Problem, Change, and Configuration Management is likely to be part of any ITIL implementation, the services that fall under each category, the relationship of the Service Catalog to these various processes, the ways requests are structured and managed and many other details will vary widely based on the needs and preferences of a particular organization, as well as the functionality of the particular software program used to implement ITIL.

Companies may implement ITIL along a spectrum moving from simple to complex – from a fairly streamlined, direct approach, to a more controlled approach, with more process steps and approvals required for relatively simple tasks. Based on our years of experience implementing IT service solutions that people actually use, we have designed our out-of-the-box implementation to keep things as simple as possible within the ITIL framework.

For instance, we have chosen to put requests for standard changes that require no special approvals into the Service Request table rather than the Change request table, keeping the workflows for each request type more distinct and simple. Services such as New Employee Setup and Password Resets are therefore handled within Service Requests.

We have also designed the system so that if a person has a problem with a printer, it does not necessarily require the creation of an incident, a problem, and a change request, just to get a new ink cartridge installed. However, if you want to follow this more extended process, we have made it as easy as possible – from any record a button can be clicked to create the related records and map field values from the current record. There are no duplicate text entries when creating problems, change requests and incidents.

Since there are so many different ways of implementing ITIL, this document is a detailed guide to how it has been implemented in the out-of-the-box   AgiloftITIL KB. We have pre-built the complex relationships and functions that many companies may want, without adding too much complexity for those who prefer a less bulky and more efficient implementation. This is a rather difficult balancing act, and while we have done our best to get it right for the largest number of customers, the real power of the system is in how easy it is to change it to adapt to your company's specific preferences and needs.

We will offer some guidance throughout this document to where to go to make changes to the critical relationships and behavior to suit your needs.

General ITIL Overview

This page describes a few key features of our ITIL system.

Which ITIL Standard do we use?

Our ITIL out-of-the-box system is based on the standards and terminology defined in the ITIL 2011 Edition library.

What controls security and the items different users can see?

In general, group permissions control what each user can see and do in the system. A user may belong to more than one group to receive a superset of that group's permissions. For detailed documentation of our group permissions and how they work, click Group Permissions Help.

Can fields be designated as mandatory?

Yes, any field may be made mandatory. It can be defined as always required, never required, or conditionally required based on the value in another choice-based field. This option appears within the field setup wizard, on the Options tab:

Once fields have been designated as required, how this requirement is displayed is defined globally.

The Required Fields window, located at **Setup > Look and Feel > Required Fields Display**, is a single page that allows you to define how the system displays required fields.

The options selected in the Required Fields window apply to all users, and can be defined separately for User-record fields and for Wizard fields.

Here you can choose:

- Whether to use an asterisk (*) to denote a required field
- Where to position the asterisk
- To use alternate text defined by the administrator with custom HTML

- The color of the required label

Archiving Records

Records may be archived automatically in a variety of ways. The system can run a time based rule to set a flag in records whose Date Closed is greater than some time period, for instance 5 years, and that are not used as standard solutions or knowledgebase articles. It can set an Archived flag to Yes, and that can cause the records to be made invisible to standard users and then deleted on a regular basis, say quarterly or annually, after saving a backup with all archived records.

In our out-of-the-box setup, we have added an Archived field for Incidents with a rule to set the value to Yes three years after an incident is closed, provided it is not a standard solution or converted to a knowledgebase article.

Unique Record IDs

In Agiloft, any record is automatically assigned a unique ID number for the table in which it resides. The numbers are incremented by 1 for each record. If a customer requires a more complex numbering system, we can customize the system to add prefixes or other values in addition to the automatically incrementing number.

Audit Trails and the History Field

Each table has a History field that maintains an audit trail of every change made to the record, whether that change is made by a user or a rule action. Here is an example of a history field for an Incident:

History Events

	Date	Modified By	Modified Fields
	May 21 2016 13:50	ewsystem	Review for Knowledgebase, Date Updated, All Time Spent, Standard Solution, Total Hours to Close, Resolution, D...
	May 21 2016 13:32	ewsystem	Number of Completed Tasks, Total Tasks Cost
	May 21 2016 13:29	ewsystem	Date Updated, Show Launch Task Button
	May 21 2016 13:29	ewsystem	Date Updated, Select Tasks
	May 21 2016 13:29	ewsystem	Date Updated
	May 21 2016 13:29	ewsystem	Status, Date Updated, Show Launch Task Button
	May 21 2016 13:28	ewsystem	Date Updated
	May 21 2016 13:28	ewsystem	Number of Tasks, Show Launch Task Button, Date Updated
	May 21 2016 13:28	ewsystem	Date Updated
	May 21 2016 13:27	ewsystem	Date Updated, Select Tasks
	May 21 2016 13:22	ewsystem	**Incident Created** Priority ID, Cost Model, SLA Target Resolution Time, Priority, Assigned Team, SLA Target ID, SLA Warning Date,

Each entry may be opened for full details of the change:

May 21 2016 13:50			
Changes made by ewsystem **View Incident After Change**			
Field Changed	**From**	**To**	**By**
All Time Spent	--undefined--	0.4	
Closed By	--undefined--	Agiloft System	
Closed By Team	--undefined--	Admin Team	
Review for Knowledgebase	--undefined--	No	
Resolution	--undefined--	Completed and provided new key card to the employee.	
Standard Solution	--undefined--	No	
Date Closed	--undefined--	May 21 2016 13:50	
Working Hours to Close	--undefined--	0.514	Default
Status	In Progress	Closed	
Date Updated	May 21 2016 13:29	May 21 2016 13:50	
Total Hours to Close	--undefined--	0.514	

Rule: Edit: Edit actions (API Enabled) **View Incident After Change**

Action Name	**Action Report**		
Status Change and Notification Actions for incidents	If-then-else flow		
	search matched	Status changed from: any value to: 'Closed' during record's last modification	
	search matched	-else-	
	action executed	Set SLA Met to Yes	
	search matched	Service is Billable='Yes'	
	search matched	Cost Model='Fixed for the Service'	
	action executed	Set Total Cost from Fixed Cost	

Field Changed	**From**	**To**	**By**
Total Cost	--undefined--	$50.00	Status Change and Notification Actions for incidents
SLA Met	No	Yes	Status Change and Notification Actions for incidents

The history shows whether changes were made by a user or a rule, and also provides a button (View Incident After Change) to show a snapshot of the state of the record after this change.

Searching

Any table in the system can be searched. For instance, you may want to find a particular incident, or all contracts of a certain type, or you may be looking for particular text content within an attached file. The quick search block lets you complete *ad hoc* searches, and is used when you don't anticipate running the same search again.

Saved searches let you run the same search repeatedly and is typically reserved for searches you will need routinely. Saved searches have the added benefit of accepting more complex search criteria and offer more granular search results.

Quick Searches

Quick searches are done from the search block above the table you wish to search. If the search block (screenshot below) is not visible, click the word Search on the action bar to expand the search block:

Using the Search Block

- The first input box is a drop-down to indicate which field in the table to search. If the default -TEXT- is selected, all fields will be searched, including the content within any attached files, including .doc, .docx, text, and CSV files as well as OCR'd PDF files.
- The second input box defines the search operator. Different operators are available depending on the type of field selected in the first box. For example, numeric or choice fields offer "greater than" and "less than" operators, but text fields do not.
- In the third input box, enter the value you are searching for.

The most common ad hoc searches will look something like "Knowledge Type equals User Guide" or "Incident Summary contains keycard." Quick searches can be performed on related tables using the search function on the associated action bar.

Saved Searches

To create a saved search, hover over the down arrow next to the search icon and select New. This opens the saved search wizard, which has options for Simple filters (most commonly used), as well as more advanced options. Click Simple to add a filter:

Add Filters:
Simple | Time | Calendar | Advanced | Related Table | Run-Time | Duplicate | First/Last

-TEXT- | contains | Value

-TEXT-
Approval Notes
Approver(s)
Assigned Person
Assigned Team
Audience
Change Manager
Change Request Attached Files
Change Request Status
Change Request Working Notes
Change Summary
Compliance Notes
Content
Created By
Creator Login
Creator Team
Date Created
Date Published
Date Retired

Cancel

Just like the quick search above, simple filters let you search any field for a particular value. Again, selecting -TEXT- will search through all available fields within the current table. For more information on the other search types, please see our online tutorial here.

Managing Saved Searches

Users in the Admin group, and those with permission to publish saved searches, see an Apply tab in the saved search wizard where they can decide which groups should have the search:

1. Accessible
2. Active
3. Visible in left menu
4. Visible in My Assigned

The Apply tab options have the following results:
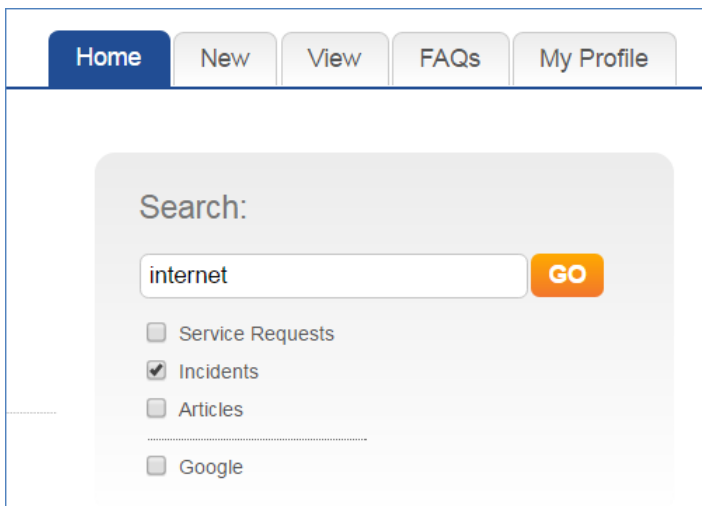
1. Choosing to make a search Accessible gives user the ability to interact with the search in their own Manage Searches wizard.

Each staff, or power user, can adjust where their accessible searches appear by checking the Active, Show in Left Pane, or Show in My Assigned boxes, defined next. This lets users customize where searches appear to tailor the interface to their work preferences.

1. Active makes the search appear on the Search drop-down:
2. Show in Left Pane adds the search to the left pane, indented below the Table name.
3. Show in My Assigned adds the search to the My Assigned section in the left pane.

Additional Searching in the EUI

In the end user interface home page, there is a search block that can be used to search tables to which the user has access. A general internal customer can search service requests, incidents, articles, or can search Google:

Groups and Teams

When you begin to customize the system, think about the different types of users and how their role affects the access they need.

Agiloft users belong to *teams* and *groups.* Groups set the level of access to tables, records, and fields. Team settings affect other parts of the interface such as the color scheme, available views, and the default home page. Teams also define working groups of users and can receive emails that go to every member of the team.

Users in multiple groups receive the superset of those groups' access settings. Users can also belong to multiple teams, but must always have a Primary Team to set important defaults. For easier maintenance, we recommend keeping the number of groups relatively small.

Terminology

A note about terminology: We use the term **end user** to mean users who access the system through the end user interface, a simplified interface that allows them to create records of any kind, view any records made available to them, edit records defined as their own, and view any FAQs made available to them. These users cannot edit records defined as belonging to other people and they use the unlimited end user license.

We use the expressions "End User" or "Customer" interchangeably in this document to refer to company employees whose main role in the system is to make requests on their own behalf or for someone else (typically their supervisor or supervisee).

We use the term **staff** to indicate the people who are working on other people's issues – they may be solvers, technical support staff, IT staff, approvers, developers, sales reps, managers, or any other types of users who access the system through the staff interface.

"Technician" may also be used to refer to members of the IT organization or other teams that will be responsible for handling, creating, or responding to requests submitted by customers or other technicians.

Both end users and staff users may be employees of your company. Staff users require their own named license or may share a concurrent staff license.

- Groups
- Teams

Groups

This table lists the default groups that have been set up for the ITIL KB and describes the general permissions of each. To see more information about a group's permissions, you can print out or save to a file the full details of that group's permission by going to Setup/Access/Manage Groups, selecting the group, and mousing over the printer icon to choose the printout you want.

Groups	Type	General description of their access permissions
Admin	Staff	These users can edit the system including rules and workflow, and can view, change and delete all records
Approver	Staff	This group holds people who can approve Contracts or Change Requests. Approvers will primarily interact with their Approval records, but they can also view change requests for which they are an approver. They may be added to the Change Approval Board or the Emergency Change Approval Board Teams. They can also view and edit Contracts for which they are an approver, and can view tables necessary to approve Contracts such as Steps, Approvals, Approval Templates, and Companies.
Base ServiceDesk	Staff	This group has the base permissions that should apply to all more privileged groups dealing with the Service desk tables. Users in those groups should also be made a member of the Base Servicedesk group. This group has full create/edit access to records in the Service Request, Incident, Problems, and Tasks tables, and create/edit own access to Change Requests and time entries. It has full view access to Configuration Items, Services, Companies, and Employees and can edit its own employee record, but has no other create or edit access in those tables. It can create/edit end users records, but not delete records.
Document Creator	End user	Can create documents and edit their own – customer of document table.
Document Manager	Staff	People who can approve and publish documents. Cannot modify status of document records manually.
Document Reviewer	Staff	Can edit approvals for which they are the approver.
Service Manager	Staff	For staff responsible for maintaining the Service Portfolio (Service table) and the Task Workflows/Templates table. Only the Service Manager group can create new Services.
Change Manager	Staff	This group is responsible for management of Change Request records and has full privileges on the Change Request table. Members can create, edit, and delete records in this table and will typically be Change Managers or Change Owners. They can also create task and approval workflows for Change Requests, and can edit Change Request related services. Change Managers will typically also be on the Change Advisory Board and possibly the Emergency Change Advisory Board Teams.

Configuration Manager	Staff	This group has full access to the Configuration Item records and is responsible for creating, editing, and deleting those records. People responsible for working on and configuring CIs, managing CI resources, and so on would typically be in this group. They might also be added to the Service Manager group if they are responsible for setting up change request workflows or services related to CI's.
Procurement Group	Staff	This group is responsible for managing the Purchase Request and Item tables.
Internal customer	End user	Internal Customer in employee table, can request Service Requests, Purchase Requests, and report Incidents, as well as see their own Configuration Items and edit some of their profile information and view other employee contact information. They may also access Knowledge FAQs.
Contract Creator	End user	For internal employees who can create contracts.
Contract Manager	Staff	This group has full access to the Contract table, Approvals table, Steps and Approval Workflow tables, and Companies table. They also have some access to End Users and Employees. They are responsible for creating, editing, and approving contracts for customers or the company.
Contract Owner	Staff	This group has a subset of the permissions Contract Managers do. They are responsible for Contracts assigned to them, and have full permissions there, but can only view Contracts that they did not create or were not assigned to. They have all the other permissions necessary to allow them to use Contracts effectively.
Customer Manager	Staff	Customer Manager – relevant if providing external customer support. Customer managers can view all support cases for their own company.
Customer	End user	Unused unless providing external customer support. Then this group is used for end user customers, who can submit and view their own support cases.
Sales	Staff	This group is responsible for recording information regarding sales efforts to specific companies as well as Purchase Orders made. They can also create and update Support Case records for the companies they represent. They have full access to: Company, Contract, Lead, Opportunity, and PO tables. Partial access to Campaign, Product, Product Quoted, Project, Quote, Support Case, Tasks, Teams, Time Entry, People: End Users.
Marketing	Staff	This group is responsible for coordinating and recording information about marketing campaigns and providing quotes to prospective customers. They have full access to: Campaign, Company, Lead, Opportunity, and Product tables. They have some access to Product Quoted, Quote, Tasks, Teams, Time Entry, People: End User, and People: Employee tables.
Guest	End user	This group may be used in hyperlinks to allow creation of new requests of any kind, such as Leads, users, Incidents, without seeing the rest of the user interface

| Anonymous | Staff | This group is used to enable unregistered users to click on an email hyperlink sent in an outbound email in order to edit that record. It is used in conjunction with the Anonymous user. If all your users reside in the system, you will not need this group. |

Teams

Teams serve a different purpose for customers/end users and for staff users. For end users, unless you need to provide multiple branded interfaces to different sets of users, or expect your end users to access the system in multiple languages, it is simplest to put them all in one team with the word 'customer' in the title, such as Customer Team or Internal Customer Team.

If you need to provide a custom look and feel and customer branding to different sets of users, then you will need a separate customer team to go with each look and feel scheme. Teams can also be associated with a different default language, so it would make sense to have language-based customer teams if you were planning to run the program in a multi-lingual environment.

For staff users, teams are used to identify the functional units to whom records may be assigned. You will want a staff team for each assignment group, sometimes called a Queue. Teams can be hierarchical, so you may set up a hierarchy such that you can send an email to a mid-level team and the members of its subteams will receive the email. Users have one primary team, which defines the look and feel they see and their default table views. They can be members of as many additional teams as needed, so they are CC'd on emails and included in the assignment list for items assignable to those teams.

Teams Table is Special

The Teams table is a system table with some special hard-coded fields, such as the Working Hours and language fields, the team name, label, description, Team Leader, and some screen refresh options.

It also has some custom fields that can be modified, and you can add your own custom fields to the teams table to manage any other information associated with your teams. If you edit the teams table setup, you will only be able to modify the custom fields and their layout options.

Creating new teams and accessing all of their attributes is done through the **Setup > Access > Manage Teams** screen. Editing just the custom fields and adding users may also be done through the Teams tab on the left toolbar.

Managing Team Membership

Team membership is controlled by two fields in the user record that are linked to the Teams Table: Primary Team (single choice), and Teams (multi-choice). We recommend adding a user's Primary Team to the multi-choice Teams field along with any additional teams. This way a user's entire team membership can be found in a single field, which makes searching and filtering easier.

When you initially import users into the system, you can import their primary team and teams' values, with multiple comma separated values with no spaces. Once users are in the system, you may update team membership by editing specific users or by editing the team through the Teams tab, and looking up and importing users for the embedded Primary Team Members or Team Members tables shown for the Team. Looking up and selecting a user for either of these tables changes the linked field in the user's own user record to point to this team.

Default Teams

The teams shown below have been set up as the default teams for the application. You can delete any teams you do not need and you can rename any team to match your own company's naming conventions. You can of course also create any additional teams you need. This is best done by going to **Setup > Access > Manage Teams** and either editing a team or creating a new one there.

User Teams	Description
1st Level Support Team	First Level Support Team
2nd Level Support Team	Second Level Support Team
Admin Team	Used for system notifications about rule, email and other errors
Backup and Storage Team	CI Backup and Storage Team
Change Advisory Board	This team holds all members of the Change Advisory Board. Subteams may be created for CABs for particular business units or departments as needed.
Change Approver Team	Change Approver Team
Change Management Team	Change Team
Compliance Team	Compliance Team used in contract approvals
Configuration Management Team	CI Team
Contract Management Team	Contract Management Team
Contract Owner Team	For staff contract owners/buyers
Custom Applications Team	CI Software Team

Customer Team	External Customer Team
Database Team	CI Database Team
Desktop Applications Team	CI Desktop and Support Team
Document Creator Team	Document Creator Team for internal end users. If using external users, then change the parent team.
Document Management Team	Team for Document Management
Document Reviewers Team	People who have been identified to review documents.
Emergency Change Advisory Board	This team holds members of the Emergency Change Advisory Board, which is typically a subset of the CAB Team. These members will be called upon to approve emergency changes.
Facilities Team	Facilities support team
Finance Team	Finance team used in contract approvals
HR Team	Human Resources Team. Acts on new employee Service Request Tasks.
Internal Customer Team	Internal Customer Team
Knowledge Team	Knowledge Team for publishing FAQs
Legal Team	Legal Team approves contracts
Marketing Team	Marketing Team
Network Operations Team	CI Network Team
Office Mgmt Team	Office management support team
Professional Services Team	Members of this team are assigned to projects

Project Manager Team	Project Manager Team
Purchasing Team	Purchasing Team
Risk Team	Risk team approves contracts
Sales Team	Sales Team
Security Team	Configuration Item Security and Support and Change Team
Server Team	Configuration Item Server Team
Service Management Team	This is the primary team for Service Managers
System Administration Team	Configuration Item System Administration Team
Vendor Management Team	Vendor Management Team
Vendor Team	Container for external Vendors - used in Contract Management

Special Teams to Manage Schedules and Working Hours

Since teams can have detailed working hours, we have set up a few special "teams" just to hold the sets of hours used to define a set of Support Hours, Availability Schedules, and Maintenance Window schedules. These teams do not have user members, but are linked into SLAs and Business Services and are used to measure SLA results and change request results in terms of the amounts of time services were unavailable outside their scheduled maintenance windows.

Team	Used For
Available: 24 x 7 with maintenance friday 6pm to sun 10pm	This set of hours is used as the availability schedule for moderately critical services
Available: 24 x 7 except Sat. 8pm to midnight	This set of hours is used for more critical services
Available: 6am to 7pm weekdays	This set of hours is used for less time sensitive services
Support: 7 - 7 weekdays	Used for SLA support hours for some silver or gold SLAs
Support: 8 - 5 weekdays	Used for the standard corporate SLAs

Support: 24 x 7	Used for the gold or highest level SLAs for critical services
Maintenance Window: weekdays 7pm to 6am, Sat, and Sun to midnight	Used for low criticality services as the maintenance period for changes
Maintenance Window: Friday 6pm to Sunday 10pm	Used for moderate criticality services
Maintenance Window: Saturday 8pm to midnight	Used for most time-critical services

Request Management Overview

Deciding how to segment the service catalog and what table structure to use for an ITIL implementation can be quite a challenge. The default setup of the service catalog balances the following basic goals:

- Make it easy for end users (the customers) to know where to go to request what they need and to follow up on their requests.
- Make the back-end processing and workflows simple and clear cut for the technical users and approvers who must process the requests.
- Make the system easy for administrators to understand and maintain so that changes can be implemented for one kind of request without affecting how other records are processed.

The Service Catalog offers services to the end user that may be created in one of two tables: Service Requests or Incidents. Whether or not a particular group of users can request particular services depends on a few things:

- Their group permissions – for instance, our default internal end users cannot directly create a Change Request, while technical users can.
- Which SLA Customer group they belong to and which business services and service offerings are included in all SLAs available to that SLA Customer group.

Internal customers will submit service requests – which can include the purchase of items – or an incident to report a problem or interruption to a business or technical service. If merited, that incident may lead to the creation of a Problem and a Change Request by technical staff.

Service Requests, Incidents, and Change Requests may include a set of tasks that need to be completed before the request can be completed. Service Requests and Change Requests can also include approval workflows prior to the work being scheduled.

The system can be designed so that end users can either…

- Create records directly in any of these tables and choose the appropriate service from the service catalog once in the record, or
- Select from a set of hotlinks to create requests for specific services without them needing to know what table will be used to store and process the requests until after they have created them.

Separating requests into distinct tables simplifies maintenance and processing compared to storing them all in a single table, since the fields, form layouts, access permissions, workflows and rules can be specific to the needs of the particular request type.

The orange arrows in the diagram above show the predefined conversion paths a request may easily take, if a technician believes it is appropriate: a service request can, with a click of a button, generate an Incident or a Change Request, or can be linked to existing incidents or changes. Likewise, an Incident can generate or link to an existing Problem, while a Problem can generate or link to a Change Request.

A Problem may be related to multiple Incidents and a Change Request may be related to multiple Problems, Service Requests, and, through Problems, Incidents. Naturally there are relationships within each of these tables to Users, Teams, Configuration Items, and other essential components of the system.

Service Requests vs. Change Requests

For some services, it may be unclear whether they should be offered through a service request or a change request. For instance, account provisioning, employee off-boarding or on-boarding, or simple application changes could go under either category. We like to use the notion that changes that do not require approval may best be handled in the service request table, while those that do require changes or require careful scheduling belong in the change request table. Approvals are available to both Service Requests and Change Requests, so the services can be configured accordingly.

SLAs are primarily tied to service requests and incidents, with less weight given to SLA targets for change requests, since changes must often be carefully scheduled during maintenance windows. Thus if you are concerned with measuring turnaround time for standard services, they are generally best handled in service requests.

Service Requests Table

The Service Requests table holds all customer service requests. A service request (SR) is created when an internal customer needs a new service, a standard (preapproved) change to an existing service, a password reset, or some other kind of assistance or information. Service requests are also used to order equipment, supplies, or new services or software.

When the user needs to report an interruption in service or a problem with an existing service, an Incident should be created instead. We have chosen to separate service requests from incidents because these two kinds of request have rather different workflows and resolution criteria. An incident may often require the generation of a Problem and a Change Request, while Service Requests will generally be self-sufficient and relatively quick to resolve.

The easy way to distinguish Service Requests from Incidents is to recognize that Service Requests are the kinds of things you will choose from a service catalog, such as application assistance, password resets, new employee setup, documentation or copying services, and so on.

If a user creates a Service Request when it should have been an Incident, the technician may click the Save Changes and Copy to Incident button on the Related Records tab of the SR form to easily convert that Service Request into an Incident and have all the fields mapped appropriately.

A user creating a service request chooses the business service from a drop-down, and then sees all the active services for that business service in a second drop-down list. The default business services for service requests can be seen in the business services table, or by creating a SR record and opening the drop-down list.

Creating Service Requests

Service Requests (SRs) may be created by telephone support staff or technicians on behalf of customers or by customers directly through the web interface or by inbound email, should an inbound email account be set up.

End User Record Submission

A customer can submit a service request by either:

1. Hovering over New and selecting Service Request from the home page.
    a. When a service request is created in this way, the user is required to select a service after selecting a business service. When the service is selected, the user will see a description of the service, any special instructions for that service, and any additional dependent fields defined to be visible for that service.

2. Clicking on View My Service Catalog under the Service Catalog section on the home page.
   a. This brings the user to a list of business services with descriptions.

   b. After selecting one of the business services, the user can choose from a list of relevant services. The services shown, in addition to being filtered to services for the business service item selected, are also limited based on the user's SLA Customer Group. For instance, the user in this example is in the Sales Department SLA Customer Group, and therefore can only view services in which the Available SLA Customer Groups contains the Sales Department.

3. The Create Request button can be used to generate a new Service Request or Incident, depending on the Request Type defined for the service.
   a. If the user clicks Create Request for Set up Email Account, a new service request record opens with the service information prepopulated – the first categorization chosen maps into the Business Service, and the service offering selected maps into the Service.

When a customer submits a Service Request using either method, the contact information fields automatically populate based on the details in his/her user record. In the default setup, a request cannot be submitted on behalf of a user who has no user record in the system. This may easily be changed by modifying the linked fields from the User table to enable "non-source" values to be used.

Requests involving Purchase

1. When a user submits a request that involves a purchase, they will see the Items Requested section at the bottom of the form. To choose the items for the purchase, click the Select Item link to bring up a new item requested form.

2. The user must select the Type and Subtype – only those values made accessible in the service record will be selectable – and then look up the available models from the Catalog - shown as the drop-down called Item Name/Number - and select one.

3. Once the item is selected, the latest price will be shown and an estimated cost displayed.

4. When the user saves the Item Requested, they can repeat the process if the purchase involves more than one item; for instance, for a new computer, the user could create items for the main processor, the mouse, and the monitor to be ordered. Once the items have been created, they will be shown in the screen.

5. When finished filling out the service request, the user must click Save to save the form, after which they will receive an email acknowledging receipt of the request.

Following Progress on Their Requests

Users may return to the portal to check on the status of their service requests by clicking on View My Service Requests on the home page.
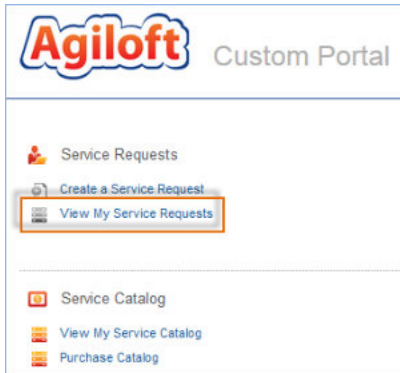
They will see all of their Service Requests displayed, and can click the edit icon to add more information to them:

Technician Record Submission

If a technician creates the SR, he may choose an assigned team and assigned person from the drop-down fields, and his changes will not be overwritten – the rule only assigns if the Assigned Team field is empty when the record is saved.

New SRs are created in a Status of Open by default. During creation, a technician may change the Status to In Progress – if they wish to indicate they are already starting work on it – or Closed – if they were able to resolve the issue immediately and want to capture the request for reporting purposes.

Automatic Emails Sent upon Submission

If a Support Staff technician creates a record in a status of Closed an email is sent to the customer telling them how to reopen the Service Request. This email is sent by a workflow action and is displayed as a checkbox that can be turned off by technician users. This option is set in the workflow options, and can be modified. If the status is not Closed when saved, the rule named All New Service Request Actions will send the customer an acknowledgement email and will also send an email to either the Assigned Person, if there is one, or to the Assigned Team.

Service Request Fields

This section describes the information that can be stored for service requests.

Common Area

The common area, shown within all tabs of a service request record, contains the ID, Summary, Status, Assigned Team, Assigned Person, and SLA Due Date for the service request. Summary is required and allows the user to briefly describe the request. Status defaults to Open and changes as the service request progresses to Pending Approval, Approved, In Progress, Pending Customer, Updated by Customer, Closed, Canceled, and Reopened. The Assigned Team and Assigned Person can be entered upon creation manually by a technician; otherwise a rule sets the Assigned Team automatically upon saving the new service request. Once the service request is saved and reopened, the SLA Due Date will be populated. In addition, the common area has a button to Cancel the Request.

Details Tab

The Details tab contains information about the user who submitted the request, the service being requested, the request's importance, and notes about the request. The Submitter fields default to the user who created the service request, but can be updated by a technician if the request was created on behalf of another user. Upon submission, the user must choose a Business Service. Based on the Business Service chosen, a list of relevant Services will be available for selection. Alternatively, if the submitter requested the service through the service catalog, these fields will be prepopulated for them.

Impact and Urgency are also required. Upon saving the record, a Priority will be chosen automatically based on the Impact and Urgency. If the Impact or Urgency is changed, the button Set Priority can be used to refresh the Priority field based on the new values. To manually override the priority, Override Priority can be set to Yes. The submitter can enter more details about the request in the Description field. After submitting a service request, the submitter can enter additional notes in the Additional Information field. Any relevant files can be added to the Attached File field.

If the service selected is marked as being billable, there are two additional fields shown above Description:

*Cost Center	Allocated Con Sales ▼	*Cost Center Code	AC-7-6718-11-00

The Cost Center must be selected for a service request for a billable service. The list of cost centers available is filtered to those where the Department matches the Submitter's Department.

Approvals Tab

If the service requires approvals, fields on this tab will allow creation, editing, and launching of approval records. Depending on the approval method of the service, either Predefined Workflow Only, Manually Created in Request Only, or Both Options Available, different fields will be available. In this example, the service is Desktop Computer, which has Both Options Available for approval generation:

A workflow will be chosen automatically if the service has a Default Workflow, but can be changed manually by clicking the lookup icon. The approvals in the workflow can be generated for this service request by clicking Generate Approvals. If additional approvals need to be added, the checkbox Create Approvals in the Manual Approval Creation section can be selected. Additional details on approvals can be found in the Managing Approvals section below.

Items, Tasks, Costs, and Time Tabs

If the service for the request is marked as being billable, for instance a Desktop Phone or Headset Purchase, cost information is shown on this tab. In addition, if the service involves a purchase, the items that the user has requested will be displayed in the Items Requested for Purchase section. If the service has a task workflow, it can be launched and managed on this tab. For instance, the Computer hardware installation task workflow contains the following tasks by default:

			Task Generation					
Task Workflow Name			Computer hardware installation tasks.					
Task Workflow Description			This is used for purchase of new computer hardware or installation for in stock hardware					
Tasks								

Launch Tasks

Status: 4 record(s) found, 1 pages. Click here to re-count records.

Mass Edit | Save Changes | Cancel Changes | Remove Task - Not Needed   Views ▼ | Search ▼

☐ Edit	ID	Sequence ↑	Task Type	Task Summary	Due Date 🖉	Status 🖉	Assigned Team 🖉	Assigned Person 🖉
☐ 🗋	1580	1	Task	Unpack and test the hardware to ensure it is functional	Sep 20 2016 09:29	Assigned	System Administration Team	
☐ 🗋	1579	2	Task	Install the hardware in its new location and connect to company network	Sep 20 2016 15:29	Assigned	Configuration Management Team	
☐ 🗋	1578	3	Task	Install any necessary software and test	Sep 19 2016 09:29	Assigned	Desktop Applications Team	
☐ 🗋	**1577**	**4**	**Task**	**Update the Configuration Item record with changes**	**Sep 16 2016 13:29**	**Assigned**	**Configuration Management Team**	

| Number of Tasks | | | 4 | | | Number of Completed Tasks | | 0 |

Based on the Service selected and the Assigned Team for the Tasks, each Task generated in this tab will also be tied to an OLA Target, with a Resolution Time and Resolution Time warning threshold, allowing progress to be tracked against them as needed and notifying the appropriate parties - i.e. Assigned Team Manager - when those thresholds are breached. For more information on OLA Targets, see Service Level Management.

The Items requested for Purchase table shows any items being purchased for this service request. Below the table is an Estimated Cost of Order based on the Estimated Cost of the Item Requested, a Total Cost of Order based on the Actual Price per Unit of the Item Requested, a count of the Number of Items Requested, and a count of the Number of Incomplete Items Requested based on the Items Requested that are not yet installed.

In the Service Cost Information, the Cost Model field is set automatically upon creation of a new service request to match the Cost Model field in the service. The options are Fixed for the Service, Based on Items Purchased, Based on Task Costs, and Based on Time Spent. Depending on the option chosen, different fields are available in the service request. For instance, if the Cost Model is Fixed for the Service, a field for Fixed Cost appears.

SLA / Work Status Tab

The SLA / Work Status tab displays all of the information from the SLA defined for the Service Request based on the user's department and SLA group. For more information on this, see Service Level Management. The SLA Title is displayed, along with the support hours for the SLA, the SLA Target Response Time, Actual Response Time, SLA Target Resolution Time, SLA Warning Date, and whether or not the SLAs for Response and Resolution were met. If the SLA was breached, a Reason for SLA Breach field becomes visible where the breach can be explained. There are also Staff Working Notes with time stamps, and a Resolution field that can pull from the Known Error database if needed along with an Attached Files field.

Related Records Tab

If the service request was created from within an incident, that incident will automatically be shown in the Spawning Incident section. A Spawning Incident can also be selected manually.

If the service request results in an incident or change request, buttons here can be used to map information from the service request into the new incident or change request. Clicking Save Changes and Copy to Incident, for example, opens a new incident record with the Summary, Submitter Assigned Team, Assigned Person,

Business Service, Description, Configuration Item, Location, Additional Information, Staff Only Notes, and Resolution prepopulated based on the service request. In addition, the incident will automatically show the originating service request in the Spawning SR fields. Once the new incident is saved, the service request will be automatically updated to show the related incident information.

If the service request is related to a release, the release can be selected in the Related Release section. Any other service requests created by the same Submitter will be shown in the All Service Requests submitted by this user section, allowing technicians to compare the current request to previous ones for the same user. At the bottom of this tab, any configuration items assigned to the submitter are listed in a table.

Processing of Records

This section describes some of the most common actions a technician will perform when working on a service request. Throughout the process, the technician can enter working notes in the Staff Only Notes field, which is an append-only field that is not visible to the customer.

Request Assignment

When a new request is saved, it is automatically assigned to a team based on the values in three fields pulled from other tables (Rule: All new Service Request actions, Action: Assign Service Request). The related service has a Responsible Team associated with it and a field called Give Service Team Priority over CI Team that has Yes/No values. If a Configuration Item was identified for the request, then there is also a CI Responsible Team field that will be populated in the SR.

The current assignment logic is: if the Service Request involves a specific configuration item (CI), then the request is assigned to the responsible team for that CI, unless the Give Service Team Priority over CI Team field is set to Yes. In this case, or If no CI is defined, then it is assigned to the Responsible Team for the Service. If there is no Responsible Team for the Service, then it is assigned to the 1st Level Support Team.

Naturally, this logic may easily be changed to suit your organization. You can simply set a default value such as the 1st Level Support Team or you may choose some other logic to use based on different fields. The current logic is implemented in the SR rule named "All New Service Request Actions" and may be modified there.

If the technician needs to reassign the Service Request to a different team or person, he or she simply changes the Assigned Team and/or Assigned Person field and the system will email the new assignee notifying them of the reassignment (Rule: Assigned team or person changed; Action: I: Notify team or person of new assignment).

Requesting Information from the Submitter

If a technician working on a Service Request needs more information from the customer in order to take further action, she can set the status to Pending Customer. A workflow action will automatically send an email to the customer requesting further information and including the content of the Additional Notes field, an append-only field that is used to communicate with the customer. The email includes a hyperlink for the customer to click to edit the Service Request directly.

Managing Approvals

For service requests that need to be approved, approvals are managed in the Approvals tab. The methods available for approval generation for a service request depend on the service.

As described in the Service Catalog and Portfolio Services, there are three options for approval generation methods for a service:

1. **Using a predefined workflow**: If a service's Approval Generation Method is Predefined Workflow Only, a button appears to Create New Approval Workflow. Clicking this button opens a new approval workflow record, where one or more approval templates can be added and sequenced. Once the new workflow saved and the Refresh button is clicked within the service, the new workflow appears as an option in the Default Approval Workflow Title dropdown. When selected as the Default Approval Workflow, the approval templates are shown in the Default Approval Workflow Templates table below.

2. **Manually creating approvals within the service request**: If Manually Created in Request Only is selected as the Approval Generation Method, no further action needs to be taken within the service. Additional fields will be available in the service request itself in order to generate approvals manually.

3. **Both methods can also be available for a service**: If Approval Generation Method is set to Both Options Available, the fields visible for the Predefined Workflow Only method are visible. In addition, for a service request created for this service, the option to manually create approvals will also be available.

If the service for the request allows for a Predefined Workflow, the Approval Workflow Title will appear and be populated with the Default Workflow for the service. It can be changed by clicking on the lookup icon. To create approvals based on the workflow...

1. Click Generate Approval(s). Any approvals generated based on the workflow will appear in the table of Approvals Needed below.

2. If the service also allows manual creation of approvals, the following section will be visible:

3. To create an approval manually, select the Create Approvals box, fill out an Approval Title, give the approval a Step Number, and click Generate Approval. An individual approval may also be selected.

4. After clicking Generate Approval, the new approval will be added to the table of Approvals Needed.

5. Once all approvals are created, click Launch Approval Process. Launching the approval process sets the approval(s) with the first Step Number from Queued to Pending Approval. In addition, an email is sent to the Approver, or, if there is no individual Approver, the entire Approval Team.

6. Approvers have three options for taking action upon approval records. They can click Approve, Require Changes, or Permanently Reject.

    a. Clicking Approve sets the Status of the approval record to Approved and triggers launching of the approval(s) in the next highest step number.

    b. Clicking Require Changes results in an email to the Assigned Person of the change request.

    c. Clicking Permanently Reject sets the Status of the approval to Permanently Rejected, sets the Status of the service request to Canceled, sets the Status of any other Queued or Pending approvals for the service request to Not Needed, and triggers an email to the Submitter and Assigned Person for the service request to notify them of the rejection.

7. Once the first step approval is approved, the next one(s) in the sequence is set to Pending Approval. Once all approvals are approved, the Status of the service request is set to Approved. SLA timeframes for the service request are applied using the date the service request was approved as the start time.

Using Tasks for a Service Request

Some services are defined to have a set of tasks that should be completed. The task method is defined in the service record as described in Tasks for Service Requests. If tasks are associated with the service, they are generated by the technician responsible for the service request when ready, on the Tasks tab. For the user-selected tasks, the Generate Tasks button remains visible in case the user decides to check additional boxes after generating the first tasks. Validation prevents the same tasks from being created a second time for the same service request.

Tasks are created with a status of Queued or Conditional (if the task is marked as a conditional task, that is, the task is only needed if some particular field condition is met, defined in the task template). A rule running on creation handles assignment. Once the tasks have been generated, the Launch Tasks button becomes visible. When the Launch Tasks button is clicked, all tasks that have no prerequisite tasks are set to a Status of Assigned, and the assigned team or person is notified with an email. The Date Due and OLA Target dates are set to start the clock from when a task is assigned. Below is an example of tasks for a service request that were generated from a task workflow in which two of the tasks have prerequisite tasks:

Edit	ID	Sequence ↑	Task Summary	Date Due	Status ✏	Assigned Team ✏	Assigned Person ✏	Prerequisite Task Titles
☐ ✏	1073	1	Disable all access	May 17 2016 11:38	Assigned	Security Team		
☐ ✏	1070	1	Conduct exit interview	May 17 2016 11:38	Assigned	Admin Team	Agiloft Admin	
☐ ✏	1072	2	Recover personal computing equipment		Queued	1st Level Support Team		Conduct exit interview
☐ ✏	1071	3	Archive all intellectual property in the user's account		Queued	Backup and Storage Team		Conduct exit interview Recover personal computing equipment

The Sequence field is used to indicate the general sequence of tasks. If all the tasks have a Sequence value of 1, this indicates that they are likely all parallel tasks triggered at the same time. If a task has one or more prerequisite tasks, its sequence value is automatically set to the highest sequence value of its prerequisite tasks plus 1. So if a task has prerequisites whose Sequence values are 1 and 4, it will be set to 5. This numbering is not functional, it is just descriptive of the general order in which the tasks are expected to be done.

If there are tasks that have prerequisite tasks, those tasks remain in the Queued or Conditional status until all of their prerequisites are completed, at which point, any conditions are checked (for the conditional tasks) and the status is set to Assigned or Not Needed (if the condition for a conditional task is not met). In the example above, the last two tasks are Queued waiting for their preceding tasks to be completed. The top two tasks were assigned at the same time, since they had no prerequisites.

When tasks are used, two fields track the number of tasks and the number of completed tasks - those with a terminal status. When these two fields match, meaning that the final task has been completed, a rule notifies the assigned person or assigned team (if no assigned person) that the final task is done (Rule: Tasks Just Completed; Action: Notify Assigned team or person of completed tasks).

Reporting Time Spent

Technicians may easily report the time they spend on handling service requests. There are two fields: Time Spent and Time Description, and an action button: Add Time, on the Working Notes tab of the layout. Entering values there will automatically create a new time entry record when the Add Time button is clicked. The time entry will show the work done by the technician and on the current date.

All the time entries for a request can be seen on the Time tab as well. If a technician needs to report time that was spent on a different day or by a different user, he may click the New button on the Time spent table to submit a time entry directly and change the date or "Done by" field. All time entered is totaled in the All Time Spent field, which can be used in reporting or billing.

The Add Time action button converts the time fields into a Time Entry record and then blanks out the current values so they will be empty. Note that this same time entry methodology is used in the Incident, Problem, Change Request, and Task tables. So time spent on each type of request will be held in a single table (Time Entries) from which reports may easily be run for all kinds of time.

Canceling Requests

At any point after creation of a service request, a user can cancel their own requests by clicking Cancel Request in the common area. Upon clicking this button, the user will first have to confirm the cancellation in a new popup window. If confirmed, the service request's Status will change to Canceled, any pending approvals for the request will be marked as Not Needed, any open tasks will be set to Not Needed, and any items requested will be marked as Canceled.

Closing Requests

When the technician has completed work on the service request, he can set the Status field to Closed and put the solution notes into the Resolution field. If a similar incident has been resolved and marked as a standard solution, a technician can click on the lookup icon next to the Resolution field:
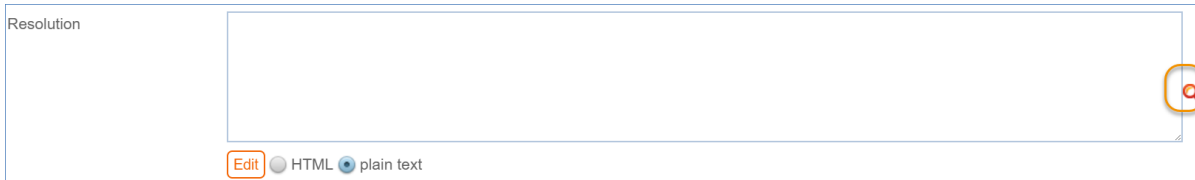
The technician can use the search bar at the top of the new window that appears to look for service requests most relevant to the one that is currently being closed. If an service request with a similar resolution is found, it can be selected and the Import/Replace button can be clicked. This will copy the service request's Resolution into the current service request, where it can be modified further if needed. The technician can also select a Closure Category, which is used for reporting. The options for Closure Category are:
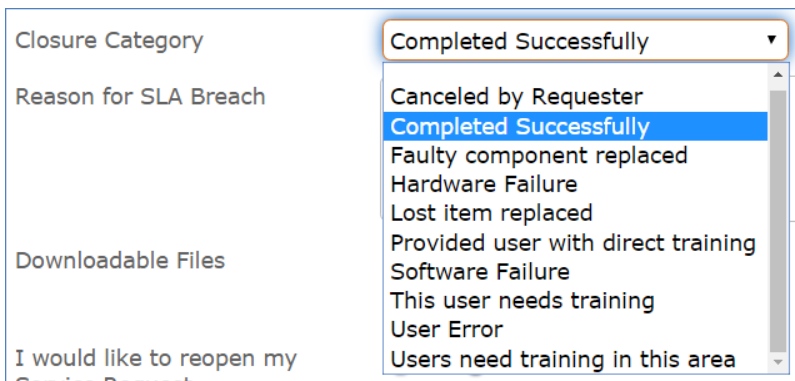
When a service request is marked as Closed, fields in the Knowledge Management section appear. Standard Solution can be set to Yes when the service request is common or has a Resolution that might apply to future service requests. Once Standard Solution is Yes, the Resolution in this service request can be selected and pulled into new service requests' Resolution fields using the lookup icon next to the field.

Setting Review for Knowlegebase? to Yes indicates that a knowledge article will be created based on this service request. Once Review for Knowledgebase? is Yes, the Knowledge Document Status and Create Knowledge Article fields are visible.The default value of Knowledge Document Status is Pending Review, and once Create Knowledge Article is clicked, the Knowledge Document Status is automatically updated to Created. If it turns out the knowledge article is not necessary, Knowledge Document Status can be changed to Not Needed manually. Clicking Create Knowledge Article opens a new knowledge article record with some information mapped from the service request, such as the Title, Purpose, and Description. Additional required fields must be populated before saving the record. Once the new knowledge article is saved, it is automatically linked within the originating service request: In addition, the knowledge article will automatically include about the source service request:

Closing a service request triggers a workflow email to the customer that includes the content of the Resolution fields and tells the customer that the request has been completed. The user can click on the first hyperlink to reopen the request. The request is opened in a new window with I would like to reopen my Service Request automatically set to Yes, and with the Additional Notes field editable. Once the record is saved, the Status is changed to Reopened. If instead the submitter chooses to submit a survey, the following window appears:

Once the survey is saved, it is added to the Surveys table, where it can be included in reports and used to inform service improvement plans.

Customer Updates

If the customer updates the Service Request at any point, an email notifies the assigned person or assigned team, if there is no assigned person, of the update.

When the customer replies to the email or edits a Service Request in a status of Pending Customer, the status changes to Updated by Customer (Rule: all customer update actions, Action: User Update Actions) and an email notifies the assigned person or the assigned team (if assigned person is empty) that the customer has replied. The customer is able to update the Additional Notes field directly and email replies are mapped to that same field.

The closing email to customers includes a hotlink back to the record if they wish to reopen it and instructs them to explain why they are not satisfied with the solution. Clicking the hotlink will automatically change the "I Would Like To Reopen My Service Request" field to Yes, which in turn sets the Status of the Service Request to Reopened and notifies the assigned person (Rule: All customer update actions, Action: User Update Actions).
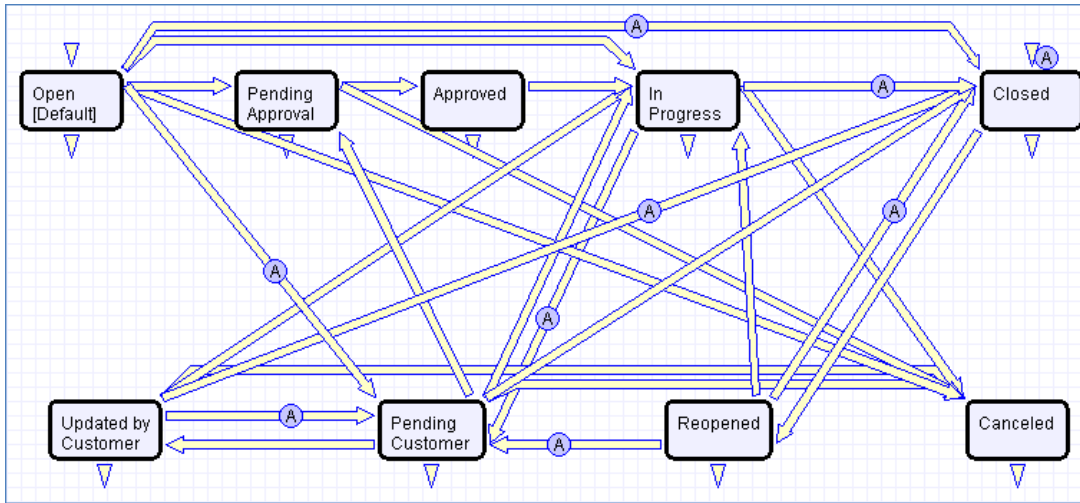
Workflow

Ownership

Records in this table are "owned" by the individual customer. This means each record is associated with a particular login and no other "end user" employee will be able to edit that record. All technician users are able to edit any service request by default.

Reporting and Statistics

Several fields are included that are used for statistical reporting. The following fields may be of interest:

- **Number of Assignees** – is auto-incremented each time the assigned person changes (Rule: Assigned team or person changed, Action: Notify team or person of new assignment)
- **Number of Teams Assigned** – is incremented each time a different team is put in the Assigned Team field (Rule: Assigned team or person changed, Action: Notify team or person of new assignment)
- **Number of Reopens** – is incremented each time a closed request is reopened by a customer
- **Total Hours to Close** – elapsed time between Date Created and Date Closed (default value)
- **Working Hours to Close** – elapsed time between Date Created and Date Closed minus the non-working hours of the team in the Assigned Team field and the time during which the Status was Pending Customer (default value)
- **Solved Within SLA** – yes/no field with default value of Yes, is set to No when the total hours to close is greater than the relevant SLA Hours to complete field, pulled in with the service and based on priority. Rule: Status Change Actions, Action: All status change actions

There are default reports measuring Total Time Spent by Service Category, Average time to close by service category and by person who closed, as well as averages for number of people assigned, number of teams

assigned, number of reopens, and so on. With the field structures already there, it is easy to add reports to slice and dice the information the way you need it. Below are a few examples of some default reports and screenshots:

Service Demand Report

This is a Trend graph that shows a trend analysis of the service demand over time based on the number of Service Requests for each Service.
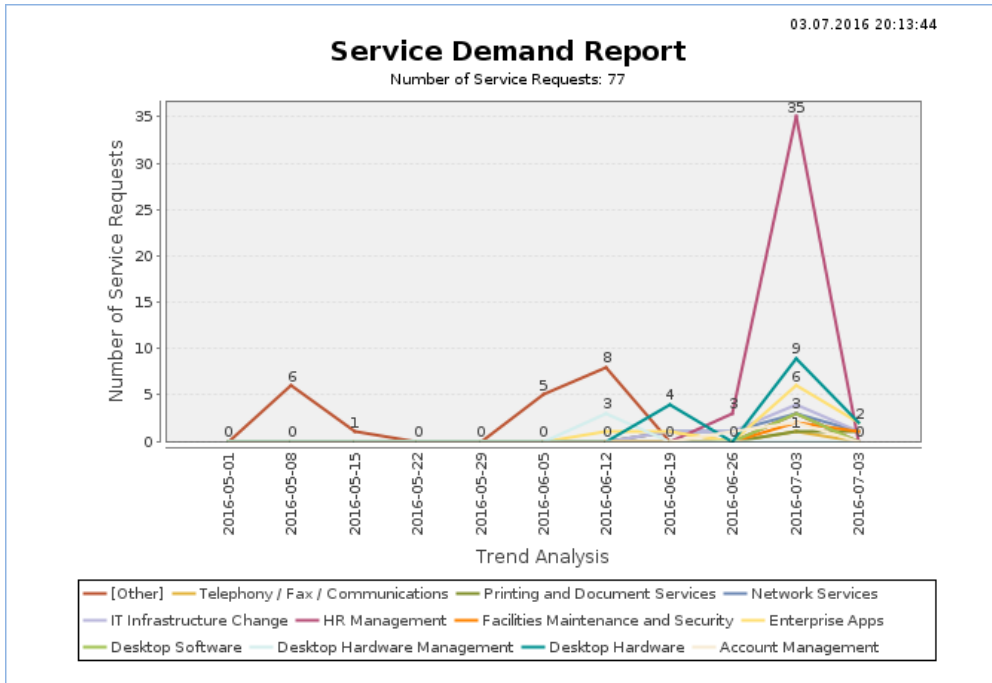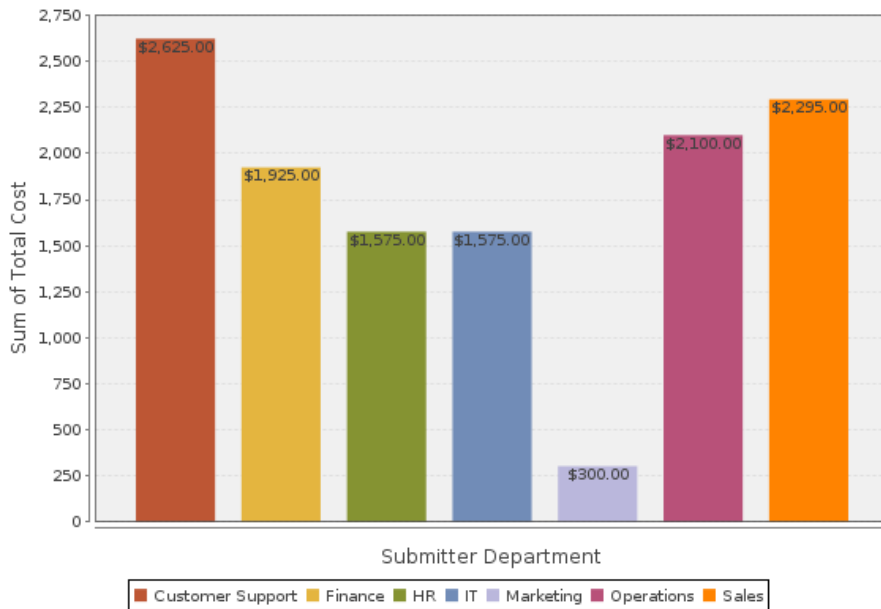

Billable Service Report

This is a Bar Chart that shows the dollar amount of Billable Services provided for each Department.

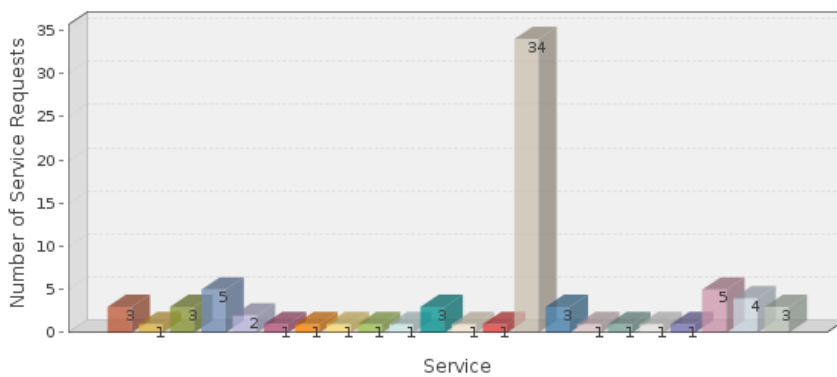## Active Service Requests Segmented by Service

This is a Bar Chart that shows the number of active Service Requests for each Service.

Automation

The following rules run in the Service Requests table. Each of them runs when a record is created or edited, or on a scheduled basis.

Create 1: First actions: Set Priority, SLA and Assigned Team and Cost Information

Rule Trigger: This rule runs whenever an incident is created.

Description: Whenever a new service request is created, the appropriate SLA is automatically set either to the SLA assigned to the CI or, if there is no CI for the incident, an SLA is chosen based on the incident's service the submitter's SLA groups.

If the Assigned Team was not selected manually during creation, it is assigned based on the following logic: if there is a CI for the incident and the service does not specify to give priority to the service's Responsible Team over the CI Team, assign the incident to the CI Team. Otherwise, if there is a Responsible Team for the service, assign it to that team, If not, set the Assigned Team to the 1st Level Support Team. If the service is billable, update the service request's Cost Model, Fixed Cost, and Hourly Rate based on values in the service record.

Set the Priority based on the Impact, Urgency, and service Priority Group.

```
if (CI SLA ID!=NULL) {
        Set the SLA to the Configuration Item SLA
} else {
        Set the SLA to the Service, Customer or Corporate SLA
}
if (Assigned Team=NULL) {
        if (Give Service Team Priority over CI Team!='Yes') {
                if (Item Responsible Team!=NULL) {
                        Set assigned team to ci responsible team
                } else if (Application Responsible Team!=NULL) {
                        Set Assigned Team from Application Responsible Team
                } else {
                        Set assigned team to 1st level support team
                }
        } else if (Service Responsible Team!=NULL) {
                Set Assigned team to service responsible team
        } else {
                Set assigned team to 1st level support team
        }
}
if (Service is Billable='Yes') {
        Set Cost Fields from Service
}
Set Priority Value
```

Edit: All Customer Update Actions

Rule Trigger: This rule runs whenever an incident is edited by the Submitter who is on a customer team.

Description: When the submitter sets I would like to reopen my Service Request to Yes when the current Status of the request is Closed, change the Status to Reopened and increment the Number of Reopens by 1. If the submitter updates the record when the current Status is Pending Customer, change the Status to Updated by Customer. If the Assigned Person field has a value, email that person to notify them that the submitter has updated the request. Otherwise, if the Assigned Person field is empty, email the entire Assigned Team about the submitter update.

```
if (Status='Closed' and I would like to reopen my Service Request='Yes') {
        Change Closed to Reopened
        Update # of time reopened field
} else if (Status='Pending Customer') {
        Change to Updated by Customer
}
if (Assigned Person!=NULL) {
        Email assignee customer update
} else {
        Email Assigned Team of customer update
}
```

Edit: Status Change Actions

Rule Trigger: This rule runs whenever the Status of a service request is changed by a user.

Description: When the Status changes to Closed, the I would like to reopen my Service Request field is set to No. If the Date Closed is also greater than the SLA Due Date, the SLA Resolution Met field is set to No and the user is required to enter notes in the Reason for SLA Breach field. If the Date Closed is not greater than the SLA Due Date, the SLA Resolution Met field is set to Yes. If the Assigned Person field is empty, it is set to whoever closed the record. If the Service is Billable field is Yes, the Total Cost is calculated based on the Cost Model. The SLA calculations within the SLA (Open Incidents - Breached SLA, Open SRs - Breached SLA, Number of Open Incidents, and Number of Open Requests) are also refreshed and the Alert Color is set to the default value.

If the Status changed to Canceled, the Alert Color is set to the default value.

```
if (Status changed from: any value to: 'Closed' last user's modification) {
        Set I would like to Reopen my SR to No
        if (Date Closed>SLA Due Date) {
                Set SLA Resolution Met to No
                if (Reason for SLA Breach=NULL) {
                        Must Provide SLA Breach Reason
                }
        } else {
                Set SLA Resolution Met to Yes
        }
        if (I would like to reopen my Service Request='Yes') {
                Set I would like to Reopen my SR to No
        }
        if (Assigned Person=NULL) {
                Set Assigned Person to person who closed
        }
        if (Service is Billable='Yes') {
                if (Cost Model='Fixed for the Service') {
                        Set Total Cost from Fixed Cost
                } else if (Cost Model='Based on Time Spent') {
                        Set Total Cost based on Time Spent
                } else if (Cost Model='Based on Task Costs') {
                        Set Total Cost to Task Total Cost
                } else if (Cost Model='Based on Items Purchased') {
                        Set Total Cost from Total Cost of Order
                }
        }
        Refresh SLA Calculations
        Set Alert Color to Default
} else if (Status changed from: any value to: 'Canceled' last user's modification) {
        Set Alert Color to Default
}
```

TB: Check for SLA Warning and Due Dates every 15 minutes

Rule Trigger: This rule runs every 15 minutes on service requests that are not Closed or Canceled and for which the SLA Warning Date or SLA Due Date is in the next 15 minutes.

Description: If the SLA Due Date is in the past or in the next 15 minutes, the Alert Color, which sets row coloring in the incident table view, is set to Red. If in addition the service has an Escalation Team, the Assigned Team of the incident is updated to be the Service Escalation Team, and an email is sent to the new Assigned Team and Person. Otherwise, an email goes to the current Assigned Team and Person as well as the Assigned

Team Manager notifying of the SLA breach.

If the SLA Due Date is in the future and not within the next 15 minutes, and SLA Warning Date is in the past or within the next 15 minutes the Alert Color is set to Orange and an email is sent to the Assigned Person, Assigned Team, and the Service Escalation Team. If the Alert Color is not already orange, the Assigned Person is notified that the SLA Warning Date or Due Date might have been changed. If neither the SLA Warning Date or SLA Due Date is within the next 15 minutes or in the past, the Alert Color is set to default.

```
if (SLA Due Date<this minute or SLA Due Date<=15 minute in the future) {
    if (Alert Color!='Red') {
        Set Alert Color to Red
        if (Service Escalation Team!=NULL) {
            Set Assigned Team to Service Escalation Team
            Email new Assigned Person and Team of SLA Breach
        } else {
            Email assiged person, team leader of breached sla
        }
    }
} else if (SLA Due Date>this minute) {
    if (SLA Warning Date<this minute or SLA Warning Date<15 minute in the future) {
        if (Alert Color!='Orange') {
            Set Alert Color to Orange
            Email Assigned Person, Assigned Team, and Service Escalation Team of SLA Warning
        } else {
            Email assigned person that the SLA Warning and/or Due Date have changed
        }
        if (SLA Resolution Met='No') {
            Blank SLA Resolution Met
        }
    } else if (Alert Color!='Default') {
        Set Alert Color to Default
    }
}
```

Incidents Table

Incidents are used to report service problems and outages. They are interruptions to customer service. The objective is to return to the normal service as defined in an SLA as quickly as possible with minimal business impact. An example Incident for an end user might be: "I cannot access my email".

What is the distinction between Service Request and Incident? Incidents are interruptions or degradations of an existing service that need to be rectified. They are problems that need to be solved, rather than a request for new service or a change to a functioning service, such as a new employee setup or new software application request. Incidents and Service Requests are held in separate tables.

Incidents may result from underlying problems caused by misconfiguration or hardware failures, and an underlying problem may result in the submission of several incidents by end users. When this is the case, the underlying problem may be reported and managed in a separate Problem record.
When an incident is caused by an underlying problem, the technician working on the incident can link the incident to an existing Problem record or convert it to create a new Problem record. They may also click the Save and Create Change Request if they wish. The Change Request table will keep track of Incidents that are converted onto it.

Problems may require significant changes in order to be permanently resolved. In this case, the technician working on the Problem will link it to a new or existing Change Request. While a Problem may stay open until the Change is implemented, often a workaround is found to resolve the interruption in service for the customer.

The system is set up to make it easy to generate and relate Incidents, Problems, and Changes, and to quickly and automatically propagate workarounds from problems into all related Incidents.

Creation of Incidents

Incidents may be created by internal customers through the web portal or via email, or by a technician taking a telephone call from a customer. When incidents are created through the user emailing a specific address, the subject line of the message becomes the Summary of the incident, the message body becomes the Description, and the Source field is set to Email.

Incidents may also be created by network monitoring systems configured to send problem reports to the system through one of the standard APIs. A field in the Incident form specifies the reporting source (email, web, phone, etc.). A technician may also convert a Service Request to a new Incident via an action button in the SR form if a customer submitted a Service Request when he should really have submitted an Incident. See Incident Table Rules in the Appendix for more details.

End User Record Submission

When a customer submits an Incident, the contact fields automatically populate based on the details in his/her User record. In the default setup, an Incident cannot be submitted on behalf of a user who has no user record in the system. This may easily be changed by modifying the linked fields from the User table to enable "non-source" values to be used.

If the user clicked the "Create Request" button from the view of services in the service catalog, the system will also populate the Business Service and the Service Problem Type automatically. If the user created the incident by clicking on a New Incident link, then the user will need to select the Business Service and the

Service Problem Type from the drop-down fields.

The user then describes his/her problem in more detail via the Summary and Description fields.

The user is also given two fields to describe the priority of the Incident: Impact and Urgency. Impact specifies how widespread the Incident is: A user whose desktop computer is malfunctioning would choose 'Affects Single User', whereas a critical server failure might be classified as 'Affects Company'. Urgency is a subjective measure of the criticality and time sensitivity of the Incident, and ranges from Low to Critical. Both Impact and Urgency are used to automatically determine the official Priority of the Incident, a field which is only editable by IT staff. See Incident Rules for the current logic used to set the Priority.

After providing the summary, description, category, impact, and urgency, the user clicks Finish and receives an automatic email notification when the system creates the record.

Technician Record Submission

A technician creating an Incident sees additional information relating to the status of the Incident, including assignee and status fields. Unlike end-users, staff can see an additional section on the details tab for Configuration Item Information that shows fields relating to a specific Configuration Item. By default Incidents are created with the "Problem CI Identified?" field set to No, but if a staff person determines the CI that is the root cause of the Incident, he or she may set this field to Yes and select a Configuration Item via a linked field set.

By default, Incidents are assigned based on configuration that is defined in the Service Problem Type (the service record) about which team should take priority - the CI team, if there is one, the service responsible team, or the default 1st Level Support Team. Technicians can also set the assigned team and assigned person manually via drop-down lists.

If the Relevant CI has been identified, and if the service has been defined to give the CI team priority, when the Incident is saved, a rule will assign the team responsible for the CI selected to the Incident. A technician may change the Assigned Team field later, as the rule will only set the Assigned Team at the time the Incident is first created.

New Incidents are created in a status of Open by default. During creation, a technician may change the status to Assigned (if he wishes to indicate it has already been assigned to an individual) or Closed (if he was able to resolve the issue over the phone and wants to capture the request for reporting purposes).

Automatic Emails Sent upon Submission

If a technician creates a record in a status of Closed an email is sent to the customer telling him/her how to reopen the Incident. This email is sent by a workflow action and is displayed as a checkbox that can be turned off by technician users. This option is set in the workflow options.

If the status is not Closed when saved, the rule named "Incident - All Creation Actions" will send the customer an acknowledgement email and will also send an email to either the Assigned Person, if there is one, or to the Assigned Team if not.

Incident Form Layout

This section contains an overview of the information stored in an incident record.

Common Area

In the common area, an incident Summary must be entered to briefly describe the issue. An Assigned Team and Assigned Person can be selected, and anyone who wants to be CC'd on all notifications can be added to the Internal CCs field. If an Assigned Team is chosen but an Assigned Person is not, all team members will receive a notification. If an Assigned Person is chosen, the email will be sent only to that individual.

Details Tab

The Details tab contains most of the basic information for an incident, including the submitter, service, priority, description, files, and CI.

Incident

| Save ▾ | Cancel ▾ | **Details** | Tasks / Cost / Time | SLA / Work Status | Related Records | Inactive Tasks | Emails | Stats / History | « » |

ID	90	Status	Closed ▾
*Summary	the color printer is down and no one can print		
Assigned Team	1st Level Support Team ▾	Assigned Person	▾
SLA Due Date		Internal CCs	⊙

Submitter Information

*Submitter Name	Agiloft System ⊙	*Submitter Preferred Notification Method	Email
*Submitter Phone	415-333-3333 Ext.12	*Submitter Cell Phone	
*Submitter Department			
*Submitter Service Level	Standard	*Submitter Service Catalog	Standard Service Catalog
External CCs			

Incident Information

If taking a call from a customer, change this field to Telephone

*Incident Reported via	Web ▾		
*Business Service	Printing and Document Services ▾	Service Problem Type	Other Printer/Scanner Issue ▾
Impact	Department ▾	Urgency	Medium ▾
	Set Priority	Priority	
Override Priority	◯ Yes ◉ No		

Please describe the problem in as much detail as possible providing the steps to reproduce it, if any

Description: We need this printer working, as it is our department's only color printer.

Attached Files	Attach/Manage	Solution Files	Attach/Manage
	Drag&Drop files		Drag&Drop files

Configuration Item Information

*Problem CI Identified?	◉ Yes ◯ No	*CI Type	Hardware ▾ ⊙
*CI Subtype	Network Printer ⊙		
*CI Name	Brother 2270 Printer ⊙	*CI Serial number	5468567967800687 ⊙
*CI Item Name/Number	Brother 2270 Printer ⊙		
*CI Status	Installed	*CI Team	Network Operations Team

Submitter Information

The submitter fields, Submitter Name, Preferred Notification Method, Phone, Department, Service Level, and Service Catalog in the Submitter Information section default to the person who creates the incident, but this can be changed by clicking the lookup icon.

Incident Information

The Incident Information section contains an overview of the incident itself. Incident Reported via can be used to indicate how the incident was submitted, with options of Web, Telephone, E-mail, Voicemail, and Event Monitoring System. Next, a Business Service must be selected. Depending on the Business Service chosen, certain Service Problem Types are available to categorize the incident further. Impact indicates the scope of the incident's effect, and Urgency can be used to convey how important it is to resolve the issue. Impact and Urgency have options that are available based on the Priority Group for the Service Problem Type selected. Once Impact and Urgency values are selected, the Set Priority button can be clicked to calculate which Priority to use for the incident based on the Impact and Urgency. If this automatically-generated Priority needs to be manually overwritten, Override Priority can be set to Yes, a new Priority can be selected, and a Reason for Override must be entered before clicking the Override Priority button.

*Impact	Individual ▼		*Urgency	High ▼
	Set Priority		Priority	3 - Medium
Override Priority	● Yes ○ No		New Priority Value	2 - High ▼
*Reason for Override				
	Override Priority			

The Description field allows for a more detailed explanation of the incident.

CI Information

If there is a configuration item involved with the incident, it can be selected by setting Problem CI Identified? to Yes. This will make CI fields visible, and a lookup icon can be clicked in order to search for and select the configuration item.

SLA / Work Status Tab

The SLA / Work Status tab contains information about the SLA, progress notes, resolution information, and fields to allow creation of a knowledge article. The SLA Title, SLA Support Hours, SLA Target Response Time, and SLA Target Resolution Time are set automatically based on the Impact and Urgency as well as the Priority Group of the service for this incident. SLA Response Met and SLA Resolution Met are updated automatically when the response time or resolution time has passed.

Additional Information is an append-only field showing a timestamp and author for each note added. The Staff Only Notes field also shows the timestamp and author, and is only visible to admins and servicedesk managers. The Resolution field is populated upon closing an incident, along with Closure Category. The Publication Information section appears when an incident is Closed, and contains fields that can be used when marking this incident as a Standard Solution and creating a knowledge article based on it.

Related Records Tab

The Related Records tab can be used to show records that are related to the incident. More on linking incidents to problems, service requests, and change requests can be found in the Processing Incidents section below.

Inactive Tasks Tab

Any tasks that were generated for the incident and then marked as Not Needed are shown on this tab.

Stats / History Tab

This tab shows some auto-generated dates, including Date Created, Date Updated, and Date Closed. Below, in the History Events section, each row represents a record edit made by a person or rule and shows the user or rule that made the edit, the date and time of the edit, the fields that were changed, the previous value of the fields, and the updated values.

Time Statistics						
Date Created	Jul 01 2016 09:35	Created By	Hector Gomez			
Date Updated	Jul 02 2016 17:51	Updated By	Admin JMR			
Date Closed		Closed By				
Number of Assignees	0	Number of Teams Assigned	1			
Number of Reopens	0	Priority Changed	Yes			
*Service Availability Schedule	Available: 24 x 7 except Sat. 8pm to midnight				Service Escalation Team	System Administration Team

History Events			
History			
	Date	Modified By	Modified Fields
	Jul 02 2016 17:51	devadmin	SLA Target Resolution Time Warning, Priority Override Log, SLA Target Response Time, Priority Changed, SLA Due Date, Override Priority, SLA Target Resolution Time, New Priority Value, New Priority Value ID, S... Override, Date Updated, Actual Response Time, Priority Number, Priority, Priority ID, SLA Warning Date, SLA Response Due Date
	Jul 02 2016 17:51	devadmin	Override Priority, New Priority Value ID, Updater Team, Date Updated, Updated By, Support Hours Team for SLA, New Priority Value, Updater Login, Reason for Override
	Jul 01 2016 16:54	ewsystem	Updater Login, Number of Teams Assigned, Updated By, Assigned Person ID, Date Updated, Updater Team, SLA Response Met, Actual Response Time, Status, Assignee Manager, Assigned Person
	Jul 01 2016 09:38	internal	**Incident Created** SLA Target ID, SLA Response Due Date, SLA Target Resolution Time, SLA Warning Date, SLA Target Response Time, Priority, Date Updated, SLA Due Date, SLA ID, SLA Target Resolution Time Warning, Priority Hours, SLA Title, Assigned Team

For instance, the entry on July 2, 2016 at 17:51 indicates that the user devadmin overrode the Priority field, which triggered rules to run, which further modified fields in the incident. Each line of this table can be opened to display more information.

Processing Incidents

This section describes how to process incidents.

Requesting Submitter Information

When a technician works on an Incident, if she needs more information from the customer in order to take further action, she can set the status to Pending Customer. A workflow action will automatically send an email to the submitter requesting further information and including the content of the Additional Information field, an append-only field that is used to communicate with the submitter. The email includes a hyperlink for the customer to click to edit the Incident directly.

Reassigning Incidents

If the technician needs to reassign the Incident to a different team or person, he or she simply changes the Assigned Team and/or Assigned Person field and the system will email the new assignee notifying them of the reassignment (Rule: Incident Edit Actions, Action: I: Notification Actions).
The Staff Only Notes field is an append-only field that holds technician working notes that are not visible to the customer.

Creating or Linking to Related Records

The Related Records tab allows problems, service requests, and change requests to be generated from the incident. To create a service request, for example, click Save Changes and Copy to Service Request. That will open a new service request record with some information automatically populated, including the Summary, Submitter, Business Service, and Description. Clicking Save Changes and Copy to Problem or Save Changes and Copy to Change Request will also open the new record on the screen so that it can be updated before saving, and the basic information is again mapped from the incident.

🔥 Incident		
Save ▾ **Cancel** ▾	Details	Tasks / Cost / Time	SLA / Work Status	**Related Records**	Inactive Tasks	Emails	Stats / History « »	

ID	126	Status	Open ▾
*Summary	Network issue		
Assigned Team	1st Level Support Team ▾	Assigned Person	Chris Support ▾
SLA Due Date	Jun 06 2016 17:00	Internal CCs	

Was this Incident Caused by a Change Request?

Caused by Change Request   ◯ Yes ◉ No

Related Service Request

Click this button if you want to save your changes to this Incident and map some of its fields into a new Service Request. If you want to link to a
Save changes and Copy to Service Request

Service Request ID		Service Request Status	▾
Service Request Summary			

Related Problem Record

Click this button to save your changes to this Incident and map some of its fields into a new Problem. If you want to link to an existing Problem,
Save Changes and Copy to Problem

Linked Problem ID		Problem Status	
Problem Summary			
Change Request ID			

Related Change Request

Click this button to save your changes to this Incident and map some of its fields into a new Change Request. If you want to link to an existing C
Save Changes and Copy to Change Request

CR ID		Change Request Status	
CR Urgency		CR Change Category	▾
CR Summary			

The remaining sections – Related Service Request, Related Problem Record, and Related Change Request - allow you to map information from the incident into a new service request, problem, or change request. Clicking any of the buttons will open a new record, which will contain some information auto-populated from the incident. If the incident was caused by a change request, Caused by Change Request can be set to Yes,

which makes visible a link to change requests. The source change request can be selected using the lookup. If the incident was created from within a change request by adding a new Related Incident, the Caused By Change Request field in the incident will be populated automatically.

SLAs

An SLA is assigned to each new incident automatically. If the incident has a related CI, and the CI has an SLA, that SLA is used. If not, an SLA is chosen that meets the following criteria:

1. The SLA is available for the service in the incident.
2. The SLA is available for the SLA Customer Group to which the Submitter of the SLA belongs or the SLA Type is Corporate.
3. The SLA is either Active or In Review.
4. If the previous criteria are met for more than one SLA, the SLAs are sorted by SLA Type, and the first one is chosen. SLA Types are sorted in the following order: Configuration Item, Service-specific, Customer Group, and Corporate.

Once an SLA is set for the new incident, an SLA Target is chosen. The SLA Target defines the Response Time, the Resolution Time Warning, and the Resolution Time in minutes. These times are used in the incident to set the SLA Response Due Date, SLA Warning Date, and SLA Due Date. Once these dates are defined, rules running every 10 minutes in the incident check whether the dates have arrived.

The SLA response is measured based on how long it takes a technician to change the Status of the incident. When the SLA Response Due Date arrives, the system compares the Actual Response Time to the SLA Target Response Time. If the Actual Response Time is less than the SLA Target Response Time, the system marks that the SLA Response time was met by setting SLA Response Met to Yes. If the Actual Response Time is greater than the SLA Target Response Time, or if the Actual Response Time is empty, meaning that the Status of the incident was never changed, SLA Response Met is set to No. If the SLA Response Time was not met, an email is sent to the Assigned Person, Assigned Team, and the Assigned Team Leader and a browser popup also alerts the Assigned Person and Assigned Team to the breach.

The SLA is met if the incident is marked as Closed before the SLA Due Date. If the incident has not been closed before the SLA Warning Date, an email is sent to the Assigned Person, the Assigned Team, and the Service Escalation Team to warn them of the upcoming SLA deadline. Once the SLA Due Date arrives, the field SLA Resolution Met indicates whether the incident was closed in time. If the incident is Closed before the SLA Due Date, the SLA Resolution Met field is immediately set to Yes. If on the SLA Due Date the incident is not yet Closed, the SLA Resolution Met field is set to No. In addition, if the SLA is breached and there is a Service Escalation Team, the Assigned Team for the incident is set to the Service Escalation Team and the new Assigned Team and Assigned Team Leader, in addition to the Assigned Person, receive an email alerting them of the breach. If there is no Service Escalation Team defined, an email is sent to the current Assigned Team, Assigned Person, and the Assigned Team Leader.

Tasks

Depending on the service, an incident may have predefined tasks. Within a service, there are three Task Generation Method options:

1. User Selected Tasks: Individual task templates are defined in the service without a sequence defined, and one or more of them can be selected and generated within an incident for that service.

2. Predefined Task Workflow: A series of task templates are defined in the service as the default task workflow, generally with a sequence defined. In an incident for that service, the entire task workflow can generated.

3. User Generated Ad hoc Tasks: Within an incident, tasks can be manually created and are not based on a template.

For instance, an incident with the Business Service of Facilities Maintenance and Security and Service Problem Type of Lost Keycard automatically shows three checkboxes representing options for tasks that can be selected and generated for an incident. To use one or more of these tasks for this incident...

1. Select them and click Generate Tasks. The tasks will be generated and shown in the table of Tasks below:

Tasks

Launch Tasks

Status: 3 record(s) found, 1 pages. Click here to re-count records.

| Mass Edit | Save Changes | Cancel Changes | Remove Task - Not Needed | | | | Views ▼ | Search ▼ |

☐ Edit	ID	Sequence ↑	Task Summary	Due Date ✎	Status ✎	Assigned Team ✎	Assigned Person ✎
☐ ▧	1325	1	Produce new card and update ID system to link it to the user		Queued	Security Team	
☐ ▧	1324	1	Decommission old card so it cannot be used		Queued	Security Team	
☐ ▧	1323	1	Check lost and found to see if the card was turned in		Queued	Admin Team	Admin System

| Number of Tasks | 3 **Calculate** | | | Number of Completed Tasks 0 **Calculate** |

2. By default, these user-selected tasks are parallel and can be completed in any order. If, for example, one of these tasks should only be completed only after one of the other tasks is complete, it can be edited before being launched. To indicate that the task has prerequisites, edit it and click on the Related Tasks tab.

3. Set Enable Prerequisite Tasks to Yes to make visible the fields that allow for setting up prerequisites for this task.

4. Next, select the task that should precede this one from the Add Task to Prerequisites dropdown and then click Add to Prerequisites. If there will be more than one task in the sequence number before this task, you can use the Trigger Condition field to specify whether this task should be launched when all of the tasks for the previous sequence number are complete or if any one of them is complete.

5. The prerequisite task will then be shown in the Prerequisite Tasks table and the Number of Prerequisite Tasks will be incremented.

6. Save the task. Once saved, the task's Sequence number will be automatically updated to reflect its new position in the task workflow.

7. The task chosen as a prerequisite will also be updated to show the tasks that are dependent on it.

Additional tasks can be modified in the same way. Once the task sequences are set up and any other modifications that might be needed are complete (for example, the default Assigned Team or Person could also be changed), the tasks can be launched by clicking Launch Tasks. This will mark the first task in the sequence as Assigned and trigger an email to the Assigned Person, or, if there is no individual assigned, the entire Assigned Team. In addition, the Status of the incident will automatically be set to In Progress.

If the tasks have a sequence defined by setting prerequisite tasks, tasks of a certain sequence number will only be launched once the previous tasks are complete. In our example task workflow, the two tasks with a Sequence of 2 are marked as Assigned once the first task is Completed:

Tasks

Status: Task #1323 successfully updated. 3 record(s) found, 1 pages. Click here to re-count records.

Mass Edit | Save Changes | Cancel Changes | Remove Task - Not Needed   Views ▾ | Search ▾

☐ Edit	ID	Sequence ↑	Task Summary	Due Date ✎	Status ✎	Assigned Team ✎	Assigned Person ✎
☐ ▣	1323	1	Check lost and found to see if the card was turned in	Jul 05 2016 17:00	Completed	Admin Team	Admin System
☐ ▣	1325	2	Produce new card and update ID system to link it to the user	Jul 05 2016 17:00	Assigned	Security Team	
☐ ▣	1324	2	Decommission old card so it cannot be used	Jul 05 2016 17:00	Assigned	Security Team	

Closing Incidents

When the technician has completed work on the Incident, he/she sets the Status field to Closed and puts the solution notes into the Resolution field. If a similar incident has been resolved and marked as a standard solution, a technician can click on the lookup icon next to the Resolution field. Clicking this icon opens a new window showing all previously resolved incidents marked as standard solutions:

Search again: [_____] [GO] Advanced Search
☐ String ☐ Context
Search in the Incident table and its subtables:
◉ Resolution ○ All text fields
○ Summary

Import/Replace | Import/Append | Import/Search Again | Cancel | Clear Field and Close

Incidents

🔥 Incidents
Status: 4 record(s) found, 1 pages Click for details.... 1 records are selected

☐ Edit	ID ↓	Summary	Status ✎	Priority	Assigned Team ✎	SLA Due Date	Closure Category	Business Service	Service Problem Type
☑ ▣	98	Internet connection is slow	Closed		1st Level Support Team	Jan 22 2016 15:43	Hardware Failure	Network Services	Report slow network or internet connection
☐ ▣	94	MS Word causes BSOD	Closed		1st Level Support Team				
☐ ▣	90	the color printer is down and no one can print	Closed		1st Level Support Team		Hardware Failure	Printing and Document...	Other Printer/Scanner Issue
☐ ▣	39	My whole office building can't connect to the...	Closed		1st Level Support Team		Hardware Failure	Network Services	Can't Access the Internet

The technician can use the search bar at the top of the window to look for incidents most relevant to the one that is currently being closed. If an incident with a similar resolution is found, it can be selected and the Import/Replace button can be clicked. This will copy the incident's Resolution into the current incident, where it can be modified further if needed.

Closing an incident triggers a workflow email to the customer that includes the content of the Resolution fields and tells the customer that the request has been completed. A Closure Category should also be added at this time. When an incident is marked as Closed, fields in the Publication Information section appear:

Publication Information

*Standard Solution	◉ Yes ○ No	*Review for Knowledgebase	◉ Yes ○ No
Knowledge Document Status	◉ Pending Review ○ Created ○ Not Needed	Create Knowledge Article	

Standard Solution can be set to Yes when the incident is common or has a Resolution that might apply to future incidents. Once Standard Solution is Yes, the Resolution in this incident can be selected and pulled into new incidents' Resolution fields using the lookup icon next to the field.

Setting Review for Knowlegebase to Yes indicates that a knowledge article will be created based on this incident. Once Review for Knowledgebase is Yes, the Knowledge Document Status and Create Knowledge Article fields are visible.The default value of Knowledge Document Status is Pending Review, and once Create Knowledge Article is clicked, the Knowledge Document Status is automatically updated to Created. If it turns out the knowledge article is not necessary, Knowledge Document Status can be changed to Not Needed manually. Clicking Create Knowledge Article opens a new knowledge article record with some information mapped from the incident, such as the Title, Purpose, and Description. Additional required fields must be populated before saving the record. Once the new knowledge article is saved, it is automatically linked within the originating incident.

In addition, the knowledge article will automatically include about the source incident. When an incident is closed, an email such as the following is sent to the Submitter:

If the submitter chooses to submit a survey, the following window appears:

Once the survey is saved, it is added to the Surveys table, where it can be included in reports and used to inform service improvement plans.

Cloning Incidents

If a new incident to be reported is almost identical to an existing incident, the older incident can be cloned from the main table view of incidents. To do this...

1. Select the incident to be cloned in the table view and then click Clone Incident in the action bar.
2. Clicking this button will open up a new incident record with most of the basic information pre-filled based on the cloned incident.

   Note: The submitter information is not mapped from the older incident, but instead defaults to whoever clicked the Clone Incident button.

It is also possible to copy all information from an existing incident into a new incident. We don't recommend this as it also copies the date created and other fields such as SLA due date. But if it is desirable to copy everything, you can select the incident, hover over Actions in the main table view, and click Copy.

Linking Related Incidents

When incidents are cloned or several incidents are submitted about the same problem, it is easy to link them together so they can all be dealt with en masse. Get the incidents you want on the screen, and then mouse over the Actions button to Link and then Link Selected Items:
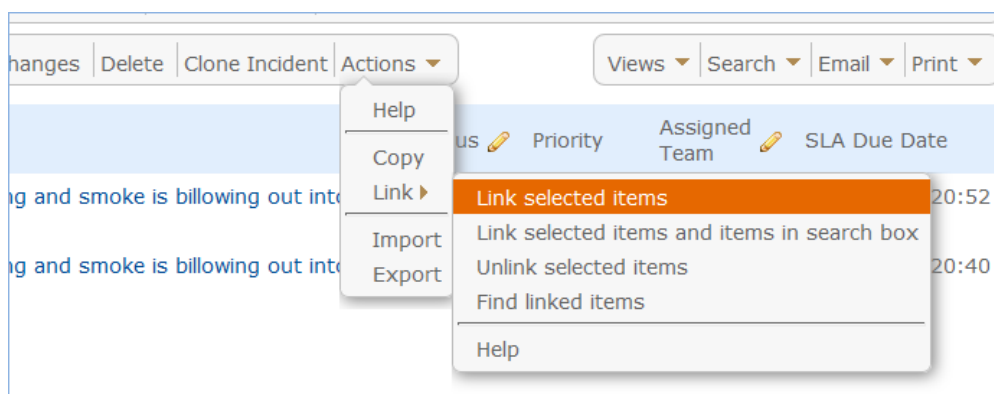
This will link each incident to the others. Later, when you want to address all of them, you can find all incidents linked to any one of them by selecting one incident and mousing over the same menu shown above and selecting Find linked items. That will bring all of the incidents on the screen. From there, you may select them all and click the Mass Edit button to close or update them all at once.

Relating Incidents to Problems

If an incident has an underlying problem that needs to be addressed, the technician may quickly convert the Incident into a related Problem record. Or she may use the Linked Problem lookup to search for existing Problems that may be the cause of this incident, and if one is found, link the Incident to that Problem. When an Incident is linked to a new or existing Problem, the Incident may be automatically updated from the Problem record when a workaround or solution to the Problem is identified.

Within the Problem record, a button may be pressed to copy the workaround for the problem into all related Incidents, set the Status of the Incident to Workaround Provided, and email the customer and assigned rep. Another button may be pressed to populate the Problem's Solution into the related Incidents, set their Status to Closed, and email the customers.

Reporting Time Spent

Technician users may easily report the time they spend on handling Incidents. There are two fields and an action button for this purpose - Time Spent, Time Description, and Add Time - on the Working Notes tab of the layout. Entering values there will automatically create a new time entry record when the Add Time button is clicked. The time entry will show the work done by the technician and the current date.

All the time entries for a request can be seen on the Time tab as well. If a technician needs to report time that was spent on a different day or by a different user, he may click the New button on the Time spent table to submit a time entry directly and change the date or "Done by" field. All time entered is totaled in the All Time Spent field, which can be used in reporting or billing.

Note that this same time entry methodology is used in the Service Request, Problem, Change Request, and Task tables.

Customer Updates

If the customer updates the Incident at any point, an email notifies the assigned person or assigned team (if no assigned person) of the update. When the customer replies to the email or edits an Incident in a status of Pending Customer, the status changes to Updated by Customer (Rule: Incident Customer Update Actions, Action: Submitter Update Actions) and an email notifies the assigned person or the assigned team, if assigned person is empty, that the customer has replied. The customer is able to update the Additional Information field directly and any email reply from the customer is mapped to that same field.

The closing email to customers includes a hotlink back to the record if they wish to reopen it and instructs them to explain why they are not satisfied with the solution. Clicking the hotlink will automatically change the "I Would Like To Reopen My Incident" field to Yes, which in turn sets the Status of the Incident to Reopened and notifies the assigned person (Rule: Incident Customer Update Actions, Action: Submitter Update Actions).

Workflow
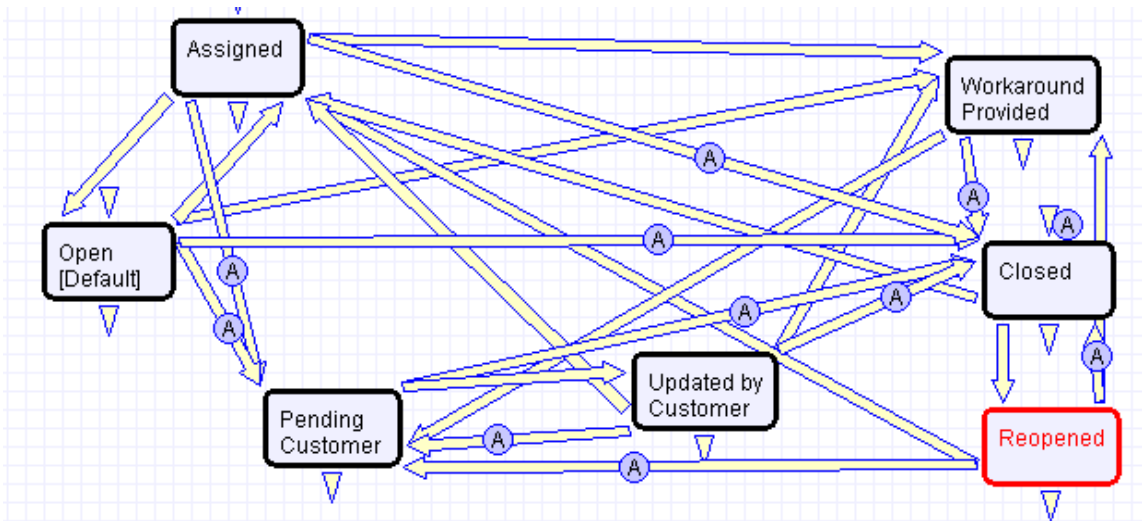
Ownership

Records in the Incident table are "owned" by the individual submitter. This means each record is associated with a particular employee login and no other "end user" employee will be able to edit that record. All technician users are able to edit any Incident by default.

Reporting and Statistics

This section describes how to create reports and gather statistics on Incidents.

Fields Used in Reports

Several fields are included that are used for statistical reporting. The following fields may be of interest:

- **Number of Assignees** – is auto-incremented each time the assigned person changes (Rule: Incident Edit Actions, Action: Notification Actions)
- **Number of Teams Assigned** – is incremented each time a different team is put in the Assigned Team field (Rule: Incident Edit Actions, Action: Notification Actions)
- **Number of Reopens** – is incremented each time a closed request is reopened by a customer
- **Total Hours to Close** – elapsed time between Date Created and Date Closed

- **Working Hours to Close** – elapsed time between Date Created and Date Closed minus the non-working hours of the team in the Assigned Team field and the time during which the Status was Pending Customer

Reports

There are default reports measuring Total Time Spent by Type of Problem, Average time to close by Type of Problem and by person who closed, as well as averages for number of people assigned, number of teams assigned, number of reopens, and so on. With the field structures already there, it is easy to add reports to slice and dice the information the way you need it. Some of the reports included are:

All Incidents updated in last 3 months by Updated By

This report contains a graphical bar chart displaying the number of incidents edited by each user. It also includes an HTML portion with more details on each incident that was edited, including the last Date Updated, grouping by user and showing the most recent update last.
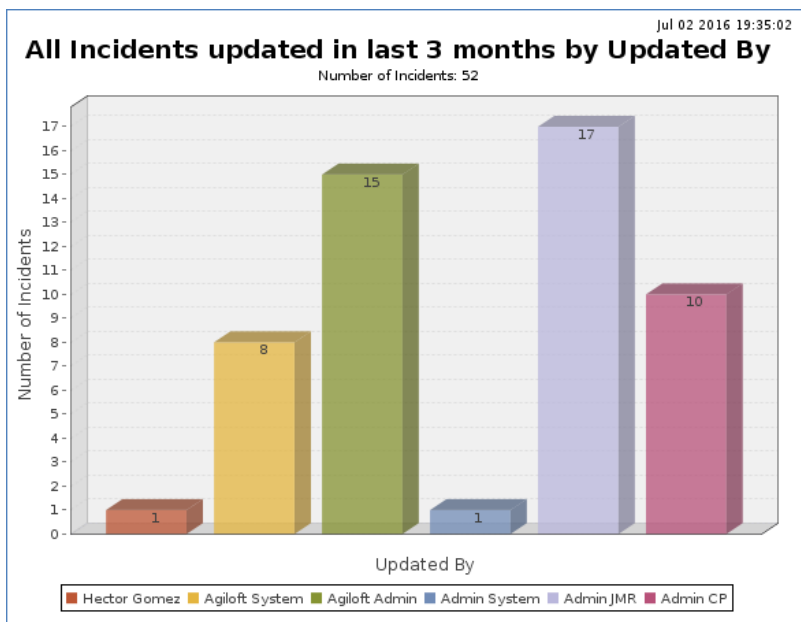
Closed Incidents, grouped by Closure Category

This report contains a graphical bar chart displaying the number of incidents closed, grouped by closure category, and also includes an HTML report.

Closed Incidents, grouped by Closure Category
Number of Incidents: 10
03.07.2016 19:18:23
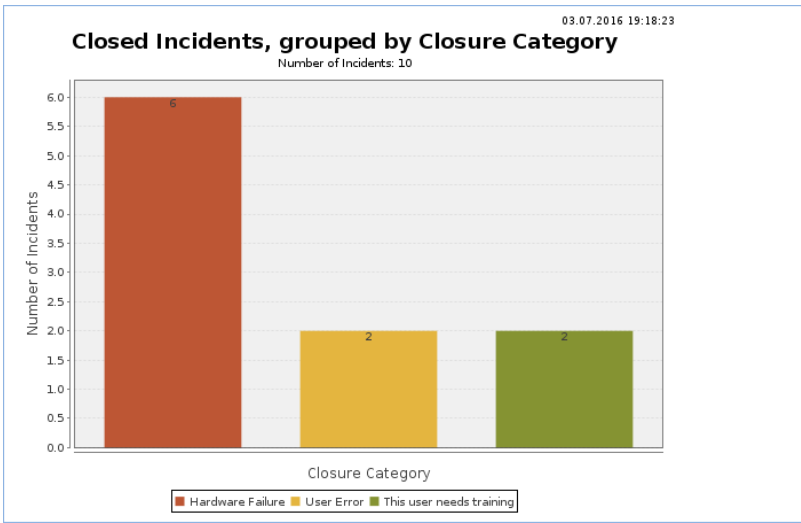
All Incidents by CI Manufacturer

This report contains a graphical bar chart displaying the number of incidents reported for each CI Manufacturer, grouped by closure category, and also includes an HTML report.

All Incidents by CI Manufacturer
Number of Incidents: 14
03.07.2016 19:22:56
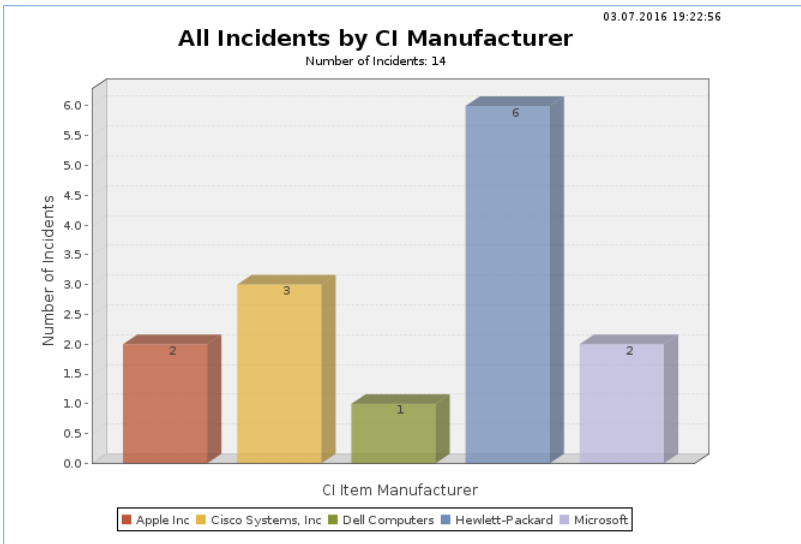
Automation

The following rules run in the Incidents table. Each of them either runs when a record is created or edited, or on a scheduled basis.

Create 1: First actions: Set Priority, SLA and Assigned Team

Rule Trigger: This rule runs whenever an incident is created.

Description: Whenever a new incident is created, the appropriate SLA is automatically set either to the SLA

assigned to the CI or, if there is no CI for the incident, an SLA is chosen based on the incident's service the submitter's SLA groups. If the new incident was created from a service request, an action updates the originating service request to add a note that the conversion was performed.

If the Assigned Team was not selected manually during creation, it is assigned based on the following logic: if there is a CI for the incident and the service does not specify to give priority to the service's Responsible Team over the CI Team, assign the incident to the CI Team. Otherwise, if there is a Responsible Team for the service, assign it to that team, If not, set the Assigned Team to the 1st Level Support Team.

If the service is billable, update the incident's Cost Model, Fixed Cost, and Hourly Rate based on values in the service record. Set the Priority based on the Impact, Urgency, and service Priority Group.

```
if (CI SLA ID!=NULL) {
    Set the SLA to the Configuration Item SLA
} else {
    Set the SLA to the Service, Customer or Corporate SLA
}
if (Spawning SR ID!=NULL) {
    Update SR with conversion note
}
if (Problem CI Identified?='Yes' and Assigned Team=NULL and CI Team!=NULL and Service Give Responsible Team Priority over CI Team='No') {
    Assign to CI Responsible team
} else if (Service Responsible Team!=NULL) {
    Assign to Service Responsible Team
} else {
    Set Assigned Team to 1st level support team
}
if (Service is Billable='Yes') {
    Set Cost Fields from Service
}
Set Priority Value
```

TB: Check for SLA Warning and Due Dates

Rule Trigger: This rule runs every 15 minutes on incident records that are not Closed and for which the SLA Warning Date or SLA Due Date is in the next 15 minutes.

Description: If the SLA Due Date is in the past or in the next 15 minutes, the Alert Color, which sets row coloring in the incident table view, is set to Red. If in addition the service has an Escalation Team, the Assigned Team of the incident is updated to be the Service Escalation Team, and an email is sent to the new Assigned Team and Person. Otherwise, an email goes to the current Assigned Team and Person as well as the Assigned Team Manager notifying of the SLA breach.

If the SLA Due Date is in the future and not within the next 15 minutes, and SLA Warning Date is in the past or within the next 15 minutse the Alert Color is set to Orange and an email is sent to the Assigned Person, Assigned Team, and the Service Escalation Team. If the Alert Color is not already orange, the Assigned Person is notified that the SLA Warning Date or Due Date might have been changed. If neither the SLA Warning Date or SLA Due Date is within the next 15 minutes or in the past, the Alert Color is set to default.

```
if (SLA Due Date<this minute or SLA Due Date<=15 minute in the future) {
    if (Alert Color!='Red') {
        Set Alert Color to Red
        if (Service Escalation Team!=NULL) {
            Set Assigned Team to Service Escalation Team
            Email new Assigned Person and Assigned Team of SLA breach
        } else {
            Email assigned person, team, and team manager of SLA breach
        }
    }
} else if (SLA Due Date>this minute) {
    if (SLA Warning Date<this minute or SLA Warning Date<15 minute in the future) {
        if (Alert Color!='Orange') {
            Set Alert Color to Orange
            Email Assigned Person, Assigned Team, and Service Escalation Team that SLA Warni
        } else {
            Email assigned person that the SLA Warning and/or Due Date have changed
        }
    } else if (Alert Color!='Default') {
        Set Alert Color to Default
    }
}
```

TB: Check for SLA Response Met every 10 minutes

Rule Trigger: This rule runs every 10 minutes on records where the SLA Response Due Date has passed.

Description: Every 10 minutes, for any records where the SLA Response Due Date has recently passed, this rule sets the SLA Response Met field to either Yes or No based on whether the response time was met. If the response time was not met, an email is sent to the Assigned Person, Assigned Team, and the Assigned Team Manager to notify them that the response time SLA was breached. In addition, a browser popup will appear for the Assigned Team and the Assigned Person if they are logged in at the time of the breach.

```
if (Actual Response Time>SLA Target Response Time or Actual Response Time='$global.NULL') {
    Set SLA Response Met to No
    Email the assigned person, team and manager of response time breach
    Remind that response is overdue
} else {
    Set SLA Response Met to Yes
}
```

TB: Check for SLA Breach - every 10 minutes

Rule Trigger: This rule runs every 10 minutes on records where the Status is not Closed, the SLA Due Date is in the past, and the SLA Resolution Met field is not No.

Description: For records found by the search, the SLA Resolution Met field is set to No. In addition, fields in the corresponding SLA and service are refreshed. In the SLA, the fields Open Incidents - Breached SLA and Number of Open Incidents are recalculated. In the service for the incident, the fields Open Incidents - Breached SLA, Open SRs - Breached SLA, Number of Open Incidents, and Number of Open Service Requests are recalculated.

U: Set SLA Resolution Met to No
L: Update SLA Calculations
L: Update Service Calculations

Problems Table

Problems are created to investigate and diagnose the root cause of one or more Incidents or possible Incidents and to decide upon a solution that will not just resolve the incident, but will prevent further incidents from being caused. Resolving a problem means resolving/preventing related Incidents.

Incident Management is concerned with restoring service as quickly as possible, whereas Problem Management is concerned with determining and eliminating root causes (and hence eliminating repeat problems).

The primary objectives of Problem Management are to prevent Incidents caused by a similar root situation from recurring and to minimize the impact of Incidents that cannot be prevented.

Problem Creation and Overview

When an incident is determined to be based on an underlying Problem, first-response support technicians create a Problem record from the Incident, automatically creating a link between the Incident record and the Problem. The new problem record imports relevant information from the Incident, such as the linked Configuration Item. Problems can also be created without being linked to existing Incidents, such as in cases where a problem is discovered internally but no Incidents have been reported.

The Submitter, along with their Department and contact information are automatically populated based on the person who creates the problem, and the reporting source of the Problem is also captured in the record. Resolution of problems may require changes to the system. Staff addressing the problem may determine that a shared CI needs to be replaced or modified, and may therefore file a Change Request. All updates throughout the life cycle of the Problem Record are logged in the record's History along with the date and time, and updates can also be logged in the Additional Notes field, which logs the updater and time stamp of the update.

When the problem is resolved, a technician updates the record with relevant information and closes the record, in turn prompting automation to begin closing procedures for related Incidents. If the problem cannot be resolved it may be classified as a Known Error and a permanent work-around supplied. This will also update the related Incidents.

Processing of Records

Once created, Problem records can be linked to Incidents from either the Problem record or from an Incident record.

Priority

Priority, a measure of the Problem's urgency and relative importance, is set by default based on the combination of Impact and Urgency and the way in which the Problem Priority group has been configured. The default priority matrix for problems is shown below:

Edit	ID	Impact	Urgency	Priority
	40	11 plus incidents	Standard	1 - Critical
	39	11 plus incidents	Emergency	1 - Critical
	42	5-10 Incidents	Standard	2 - High
	41	5-10 Incidents	Emergency	2 - High
	44	1-4 Incidents	Standard	4 - Low
	43	1-4 Incidents	Emergency	3 - Medium

These values may be changed to match a company's preferences. See the Impact, Urgency, and Priority Management section for more details. Problem Priority, a measure of impact and risk, particularly with regards to IT service operations, may be determined separately as part of Problem Classification procedures. Together Priority and Problem Priority help IT managers make factual decisions for scheduling and planning and help determine the category of related Change Requests.

Diagnosis

The Problem follows its own workflow separate from the Incident. Ops team members open the problem record in the default state of "Pending Diagnosis" to indicate that a diagnosis has yet to be determined, or "Diagnosed" if no further steps are required. The default Problem SLA has specified targets for completing diagnosis, based on the priority, and a diagnosis clock is started upon creation of the problem record. As part of the diagnosis, technicians select the service involved based on existing Incident reports or based on the most likely service to be related to the Problem. If a particular Configuration Item is identified as the source of the problem's root cause, staff can quickly link the Problem to the CI record.

Once a diagnosis is supplied, staff will move the record into a status of "Diagnosed," fill out the Root Cause description and any further analysis, and save the record. They may also suggest a temporary fix, or "workaround", for the problem and related incidents, e.g. "use Printer B3 for now instead", if a permanent solution is not readily available.

Diagnosis and Resolution Clocks

When the status is changed from Pending Diagnosis to any of Diagnosed, Pending Change, Resolved, or Deferred, the diagnosis clock is stopped, the Diagnosis End Time field is populated automatically with the current time, and the Resolution clock is started. The diagnosis clock measures the working support hours between creation of the problem and diagnosis of the problem against the SLA diagnosis target.

The resolution clock measures time between when diagnosis is completed and the problem is resolved or deferred, excluding the time when the Status is Pending Change, against the SLA resolution target for problems. While the Status is in Pending Change, both clocks are stopped, since the people handling problems may not have control over when a change request can be scheduled and implemented.

Workarounds

If there are related incidents, and there is a workaround available while the problem is being resolved, it is easy to send the workaround to all of the incident submitters. This is done by:

1. Entering the details of the workaround into the Workaround field

Workaround	Please use the network printers 432Brother in the meantime until we can get a new spare part to replace the roller.

2. Going to the Related Records tab and changing the Workaround Provided field to Yes and then clicking the button "Update Incident with Workaround". The "Update Incident with Workaround" button allows staff working the Problem to quickly disseminate workaround information to all linked Incidents with the click of a button. Clicking the button will post the workaround in all related incidents and change their status to Workaround Provided, which will also trigger an email to both the end user and the assigned staff person of the Incident(s).

Deferring and Seeking More Information

If at any time during the root cause diagnosis or determination of the proper solution staff need more information from a separate process, such as Incident details from first level support, the Problem record may be placed in a state of "Pending More Information" and an email sent to staff members with requests for further information. If a Problem is deemed too risky or of lower priority than more imminent issues, it may be put in a status of "Deferred" to reflect no ongoing diagnosis or pending changes.

Resolution

In most cases where a Problem's root cause deals with a Configuration Item, a Change Request will be submitted to make the appropriate fixes to the Configuration Item. Change Requests are creatable directly from the Problem record, instantly linking them. While a problem is waiting for a Change Request it can be put in the status of "Pending Change". If the Change Request linked to a Problem is closed, the system will send an email notification to the problem assignee so that the individual can take additional steps to close the Problem record if it was pending change for resolution.

A problem whose root cause is known but for which there is no permanent resolution is considered a Known Error. Known Errors should have Workarounds to allow Incident Management to restore service as quickly as possible. When the problem is resolved either directly, or through a change request, all related incidents can be closed at once with the click of a button. On the Related Records tab, the button "Update Incidents with Solution" can be used to push the Resolution field of the problem into the Resolution field of the linked incidents, closing the incident and triggering an email to the customer with the resolution.

Once the permanent resolution is determined and implemented, staff users enter the description in the Resolution field and set the status to Resolved. A Closure Category for the resolution of the Problem can be selected, and If the resolution contains information that is useful outside of this problem's particular scope,

the "Add to Knowledgebase?" field can be set to Yes to make the Resolution field available via FAQs. In addition, a required "Review for Knowledgebase" field is displayed in that Status, and if set to Yes, a Knowledge Article can be created from the Problem for review to be added to the Knowledge Articles table.

Analysis and Major Problem Reviews

It is important to learn from problems to reduce the likelihood of repeat issues. There are two fields intended to facilitate analysis after a major problem.

- Include in Major Problem Review: this is a required field.
- Major Problem Review notes: which used to write up notes about what was done well in the crisis, what didn't work so well, and what could be done to prevent such a crisis in the future. This field is only visible if the Include in Major Problem Review has a Yes value.

Known Errors

Problems can be marked as known errors by setting the Known Error field to a Yes value. All Known errors can be easily viewed by support technicians using the Known Errors saved search when working on incidents or service requests. Known errors that for development systems have a status called In Development, and these are shown in the search called Known Errors in Development. Only known errors for the production environment, which should be set to a status of Resolved, are shown in the Known Errors in Production search.
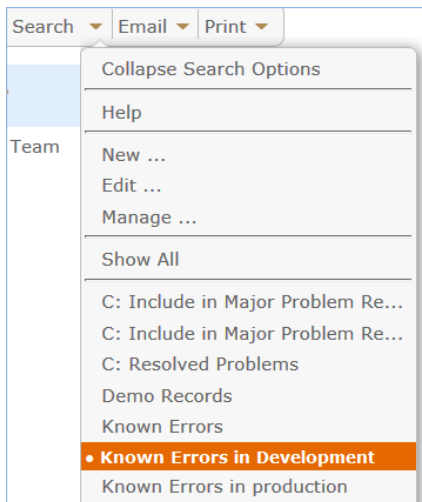
For known errors that should be visible to end users, we recommend that they are pushed into a knowledgebase article, which can be done directly from the problem record.

Ownership

Problems are "owned" by the staff member who creates the Problem record. Since only internal staff will see Problem records, groups may share responsibilities between Incident Management and Problem Management and multiple individuals may share ownership over time.

Workflow
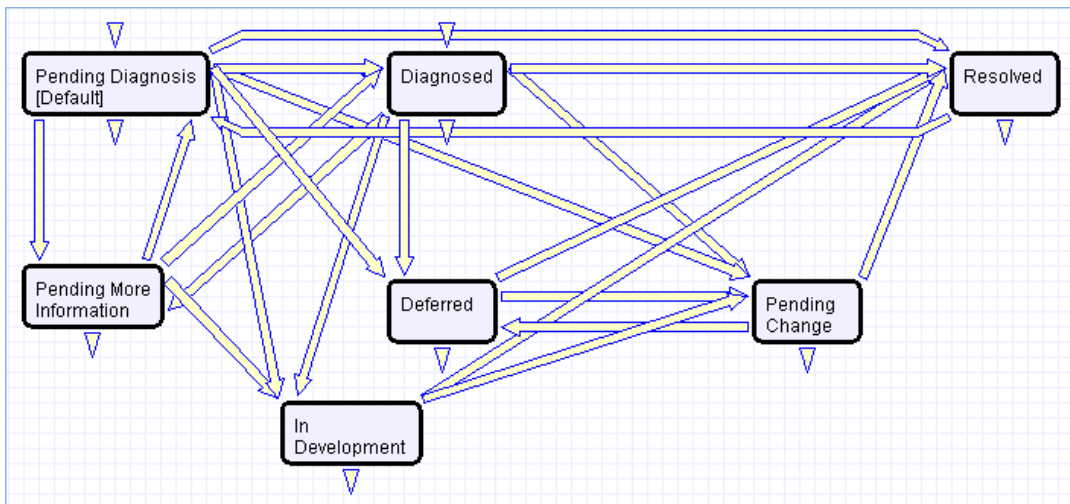
Problem Fields

This section contains an overview and screenshot examples of the information stored in a Problem record in the out of box system.

Details tab

The common area is shown in all tabs, and shows the progress of the Problem's lifecycle, such as its Status, its Team Assignment, as well as the Assigned Person or owner of the Problem record. The details tab contains most of the information and updates pertaining to the Problem, including the Submitter, Submitter Department and Contact Information, Source of the Problem (i.e. how it was reported), the Location of the problem, the Business Service and Service impacted by the Problem, as well as the Impact, Urgency, and resulting Priority of the Problem. If a CI has been identified, it can also be selected here.

In addition, all Diagnosis details can be logged here along with the Risk and Root Cause analysis, and any additional working diagnosis notes with time stamps. Finally, the Resolution section in this tab allows for the provisioning of a Workaround for the problem (which can later be pushed into the related Incidents), or to pull an existing Solution from the Known Error database. If the Status is Resolved, a Closure Category can also be specified for the Problem, and a determination can be made for if the Problem should be included in a future Major Problem Review. The date the Problem was resolved is then automatically logged in the History tab of the record to provide an audit trail.

Related Records tab

The Related Records tab shows the list of Incidents related to the Problem. If a Workaround or Solution has been provided, it can also be pushed from the Problem record to all of its related Incidents. The Problem can also be linked to an existing Change Request, or to create a new one, copying over all of the relevant

information directly from the Problem record to the Change Request. In addition, the Knowledge Management section of this tab also allows the Problem to be directly integrated and converted into a record in the Knowledge Articles table with a status of Pending Review.

SLA Tab:

The SLA tab displays the SLA thresholds for the Problem, including thresholds for the SLA Diagnosis Time, SLA Resolution Time, and the progress against those thresholds.

Automation

The following rules run in the Problems table. Each of them either runs when a record is created or edited, or on a scheduled basis.

Creation actions

Rule Trigger: When a Problem is created via Email, Web, or API.

Description: This rule runs the following If-then-else action to set the assignment of the Problem record according to the predefined values in the related Service:

```
if (Spawning Incident ID!=NULL) {
    Update "Additional Info" in Spawning Incident
    Set Source to Incident
}
if (Assigned Team=NULL) {
    if (Service Give Responsible Team Priority over CI Team='No' and CI Responsible Team!=NULL) {
        Set Assigned Team to CI Team
    } else if (Service Responsible Team!=NULL) {
        Set Assigned Team to Service Responsible Team
    } else {
        Set Assigned Team to 1st Level Support Team
    }
}
Set Priority
if (Status='Diagnosed') {
    Set Diagnosis Clock Status to Stopped
}
```

In addition, it also sets the SLA ID based on the saved search: Active, request type is problem, and SLA type is corporate, and then sets the SLA targets for the Problem based on the SLA and Priority.

Edit: Set Alert Color and Send Notifications (Web/API)

Rule Trigger: When a Problem is created edited via Web or API and meets the saved search critera: Diagnosis SLA Breached=No and Working time to diagnosis changed last modification

Description:

If then action: I: Update SLA Details

     If Diagnosis SLA Breached=No

          If Working time to Diagnosis is greater than SLA Diagnosis Warning Time and Alert Color is default

Set Alert Color to Orange

If Working time to Diagnosis is greater than SLA Diagnosis Time,

set the Diagnosis SLA Breached to Yes

Set the alert color to red.

Email the Assigned person and team of breach of diagnosis

If Resolution SLA Breached=No

If Working time to Resolution is greater than SLA Resolution Warning Time and Alert Color is default, set alert color to Orange

If Working time to Resolution is greater than SLA Resolution Time,

Set alert color to Red

Set Resolution SLA Breached to Yes

Notify the assigned person and team of breach

```
if (Diagnosis SLA Breached='No') {
    if (Working Time to Diagnose>SLA Diagnosis Time Warning) {
        Set Alert Color to Orange
    }
    if (Working Time to Diagnose>SLA Diagnosis Time) {
        Set Diagnosis SLA Breached to Yes
        Set Alert Color to Red
        Email Assigned Person and Team of Breach of Diagnosis
    }
}
if (Resolution SLA Breached='No') {
    if (Working Time to Resolution>SLA Resolution Time Warning and Alert Color='Default') {
        Set Alert Color to Orange
    }
}
if (Working Time to Resolution>SLA Resolution Time) {
    Set Alert Color to Red
    Set Resolution SLA Breached to Yes
    Notify Assigned Person and Team of Resolution SLA breach
}
```

Edit: Status Changes (Web/API)

Rule Trigger: When a Problem is edited via Web or API and Status changed during the record's last modification.

Description:

- If Status changes to Diagnosed, set the Diagnosis Clock Status to Stopped.
- If Status changes to Pending Change, set the Resolution Clock Status to Stopped
- If Status changes from pending change to some status other than Deferred or Resolve, set the Resolution Clock to Running
- If status changes to Resolved, set the Resolution Clock Status to Stopped (and if Diagnosis clock is not stopped, set it to stopped too).

```
if (Status changed from: any value to: 'Diagnosed' during record's last modification) {
        Set Diagnosis Clock Status to Stopped
}
if (Status changed from: any value to: 'Pending Change' during record's last modification) {
        Set Resolution Clock Status to Stopped
}
if (Status changed from: 'Pending Change' to: any of 'Diagnosed','Pending Change','Pending Diagnosis','Pending More Information' during record's last modification) {
        Set Resolution Clock Status to Running
}
if (Status changed from: any value to: 'Resolved' during record's last modification) {
        Set Resolution Clock Status to Stopped
        if (Diagnosis Clock Status!='Stopped') {
                Set Diagnosis Clock Status to Stopped
        }
}
```

TB: Refresh Elapsed Time fields

Rule Trigger: This rule runs every 20 minutes using the Saved Search: Diagnosis Clock Status is Running or Resolution Clock Status is Running and Date Updated is more than or = 20 minutes old (so if someone updated it in the meantime, we don't need to do it again)

Description: U: Set Date SLA Checked to NOW() - that will trigger an update of the elapsed time fields.

Reporting and Statistics

There are numerous default reports measuring different metrics for Problems, and a few of the most relevant and frequently used ones are listed here as examples:

Known Error Trend Analysis

This is a Trend graph that shows the number of Problems marked as Known Errors over the course of a pre-defined period of time (by default, this is since the beginning of the current year).

Known Error Trend Analysis
Number of Problems: 4
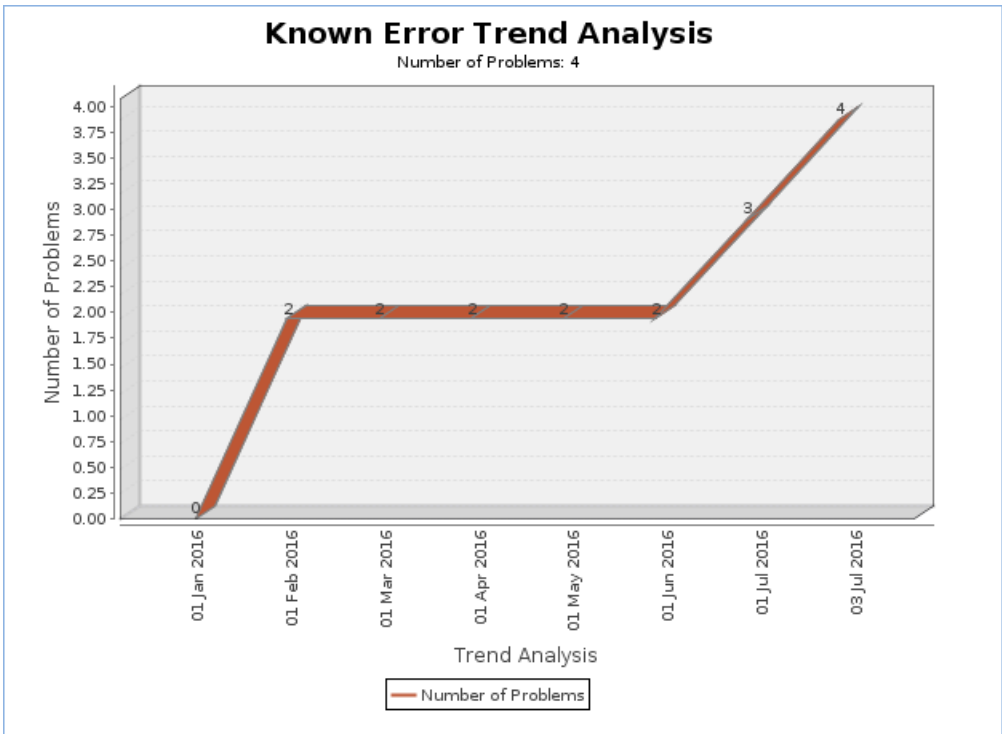
All Problems by CI Type, segmented by CI Name

This is a Segmented Bar Chart that shows the number of Problems that were reported for each CI type, segmented by the Configuration Item that they were reported for.

All Problems by CI Type, segmented by CI Name
Number of Problems: 14
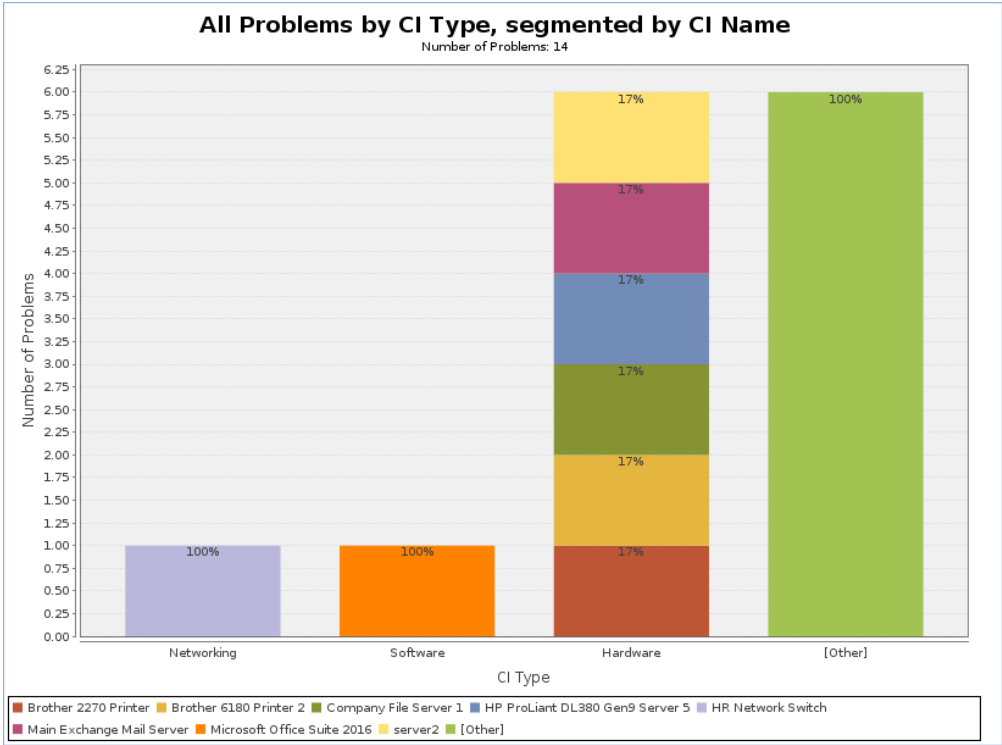
Major Problem Review Report

This report shows the major problems by month over the past year:
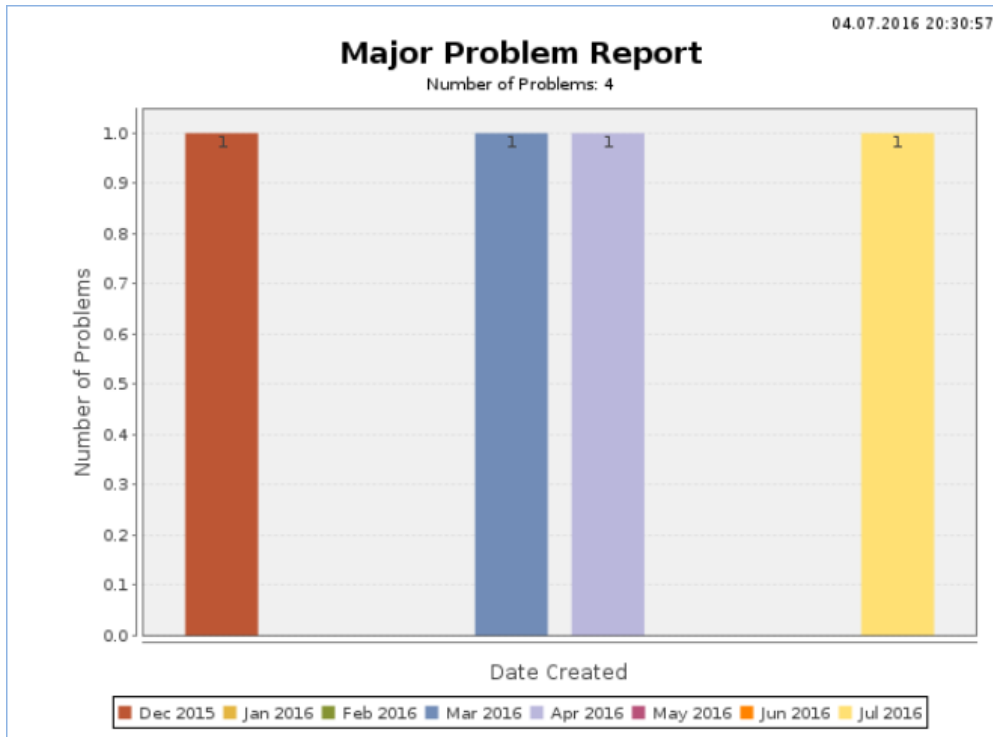
It shows further details in the HTML version of the report.

Change Requests (RFC) Table

The Change Requests table is used for Change Management. A Change request is created when a change is needed to a configuration item or to any other business object that may require a set of approvals before such a change can be completed.

Once a change request is created, it can be assigned to the appropriate teams or individuals for one or many approvals, and may then be moved along in the process, from approval to scheduling, to action by staff members actually making the change, to completion, testing, and release.

Of course each organization has its own definition of types of change that may need to be managed in this way, but we have prepopulated the service catalog with a list of common change types. You can easily add your own change categories and types to the services table. Change Requests differ from Service Requests in that they are not visible by default to end users and typically follow a stricter set of procedures and approvals.

The Change Request table pulls its services from linked entries in the Services table. A user creating a change request chooses the higher level business service from a drop-down, and then sees all the active service offerings for that business service in a second drop-down list called Service Title. The default service categories for change requests can be seen in the Configuration Items table (or by creating a CR record and opening the drop-down list). When the service is selected, several additional fields are pulled in with the service, including the number of approvals needed.

Use Case

Change Requests (CRs) may be directly created by technician users or may be created by conversion from a Service Request or a Problem, in which case they are linked back to the generating record.

End User Record Submission

By default, Change Requests are not visible to end-users and cannot be created by them. This is defined by group permissions, and if you would like to allow end users to submit change requests, you can simply change the group permissions of the relevant groups to enable this.

Technician Record Submission

When a technician submits a Change Request, the Submitter contact information fields automatically populate based on the details in his/her User record. In the default setup, a request cannot be submitted on behalf of a user who has no user record in the system. However, the creator can select a different person as the submitter if needed.

The Change Request table pulls its services from linked entries in the Services table. A user creating a change request chooses the Business Service from a drop-down, and then sees all the active services that belong to that business service in a second drop-down list called Service. The default Business Services for change requests can be seen in the Configuration Items table (or by creating a CR record and opening the drop-down list). When the service is selected, several additional fields are pulled in with the service, including fields defining whether approvals are needed and the approval generation type.Users in the appropriate groups may be given permission to change the number of approvals needed for a particular change.

When a technician creates the CR he may choose an assigned team and assigned person from the drop-down fields at the top of the screen. If he does not choose the team, when the record is saved, the assigned team will be auto-populated based on the logic described below. New CRs are created in the Status of Open by default. Once the main fields are filled in, the user navigates to the Configuration Items tab. This allows the user to look up several configuration items. The CI's can be filtered by choosing a CI Type first and then clicking the Select Configuration Items lookup.

As CIs are selected, important information can be seen in the view of the CI's, including their criticality and the diagram that shows their relationships with other CI's:

Note that if no configuration items are involved in the change request, you are still required to select one,

because of the way tasks are managed in change requests. We have created a "No CI Involved" configuration item record with the CI Type of None that can be used in this situation. This will allow all task generation and other functionality to work as expected. Once the configuration items have been selected, to further process the change request, the user needs to click the Submit for Review button: This saves the change request (validating the required fields), sets some background fields that control visibility of the approval and task buttons, and sets the status to Pending Change Manager.

Automatic Assignment of New Requests

When a new Change Request is saved, it is automatically assigned to a team based on the values in three fields pulled from other tables (Rule: CR- All Creation Actions, Action: Assign Change Request). The related service has a Responsible Team associated with it and a field called "Give Service Team Priority over CI Team" with a Yes/No value. If a Configuration Item was identified for the request, then there is also a CI Responsible Team field that will be populated in the CR.

The current assignment logic is: if the Assigned Team is empty when created and the Change Request involves a specific configuration item (CI), then the request is assigned to the responsible team for that CI, unless the "Give Service Team Priority over CI Team" is set to Yes. In this case, or If no CI is defined, then it is assigned to the Responsible team for the Service. If there is no Responsible Team for the Service, then it is assigned to the Change Management Team.
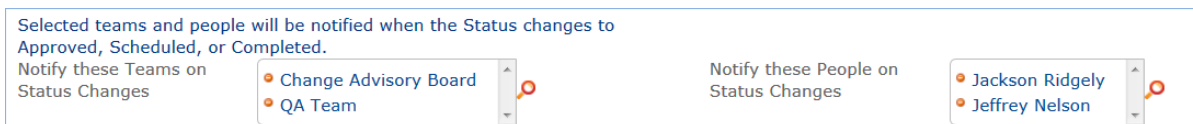
Naturally, this logic may be changed to suit your organization. You can simply set a default value such as the Change Management Team or you may choose some other logic to use based on different fields. The current logic is implemented in the CR rule named "CR: All Creation Actions" and the action named "Assign change request" and may be modified there.

Automatic Emails Sent upon Submission

If the status is not Closed when saved, the rule named "CR- All Creation Actions" will send the submitter an acknowledgement email and will also send an email to either the Assigned Person, if there is one, or to the Assigned Team.

Additional Notifications to Teams and Individuals

There are two fields that result in automatically notifying specified people or teams when the Status of the Change Request is changed to Approved, Scheduled, and Completed. These are shown near the bottom of the main Details tab:

Selected teams and people will be notified when the Status changes to Approved, Scheduled, or Completed.

| Notify these Teams on Status Changes | Change Advisory Board QA Team | | Notify these People on Status Changes | Jackson Ridgely Jeffrey Nelson | |

The change manager can look up specific teams or people and populate them into these fields, and when the Status changes to Approved, Scheduled, or Completed, the teams and/or users will receive a notification telling them of the new status and details about the request. Here is an example of the email sent when a request is scheduled:

Change request 278 has been scheduled: Upgrade RWC Headquarters connection to use Optical Carrier Transmission (OC-3)

You are identified as a cc on the change request shown below, which has just been scheduled.

The change is scheduled to start on Jul 05 2016 01:00 and to end at Jul 05 2016 03:00.

Click here to view the request directly.

ID:	278
Change Category:	Category 2 - Significant or Medium
Status:	Scheduled
Change Management Team:	Change Management Team
Change Manager:	Chuck Eisner

Processing the Change Record

Most services linked to Change Requests require one or more approvals. See the section on Approval Management for details about how to set up approval workflows and templates.

Many of the default approval workflows for change requests contain conditional approvals, which depend on the Change Type (i.e. Standard, Normal, Emergency, Change Proposal) and Change Category (Category 1 - Minor or Small, Category 2 - Significant or Medium, Category 3 - Major or Large), and other factors. The workflows are pre-defined within each service, and can be used to create Change Models and Standard Change Workflows. Each Service may have both a default Approval Workflow and a default Task Workflow.

In addition to being able to define Standard Change workflows and workflows for Change Proposals and other Change Models, further Processing Details that are pulled into the Change Request are also stored in the Service record, including the Responsible Team, Escalation Team (i.e. which team it should be escalated to if thresholds for escalation are met), the default Change Type, and any additional fields that should be shown for the Change Request based on that service.

Managing the Approval Process

Change requests can use different approval methods, discussed in the Approvals for Change Requests section on the Service Catalog page. Based on the approval method defined in the service record, approvals will typically be created and launched within the change request using buttons.

Default Conditional Approval Workflow

We have defined a standard ITIL default approval workflow called Default - conditional approvals based on type/category with conditional approvals generated based on the combination of Change Category and Change Type. It is a good illustration of how conditional approvals can be set up. It contains the following approval templates:

Step Number	Approval Title	Approval Team ✎	Assign To ✎	Approval Usage
1	Change Manager Approval for all Minor changes		Change Manager	Conditional
1	IT Executive Board Approval for Major Change	IT Executive Board		Conditional
1	CAB First Approval for Significant Normal Change	Change Advisory Board		Conditional
1	ECAB First Approval for Significant Emergency Change	Emergency Change Advisory Board		Conditional
2	CAB second approval for Major /Normal change	Change Advisory Board		Conditional
2	ECAB second approval for Major / Emergency change	Emergency Change Advisory Board		Conditional

The result of this setup is to create the following approvals for the following combinations of Change Category and Change Type:

- Minor change - Change Manager Approval only
- Significant Change / Normal - CAB Approval only
- Significant Change / Emergency - ECAB Approval only
- Major Change / Normal - IT Executive Board, then CAB Approval
- Major Change / Emergency - IT Executive Board, then ECAB Approval

Approval Creation and Processing

Once the request has been sent for review, if it was for a service that requires approvals, the change manager can review it and generate any needed approvals. Clicking the Generate Approval(s) button will generate the approvals based on the predefined approval workflow. If the change request service allows ad hoc approvals, options will be available to generate additional approvals as needed. Approvals are generated in a Status of Queued by default. Once the approvals are generated, new buttons appear:

Approvals

*Approval Workflow Title   Infrastructure Update Workflow 🔍

`Launch Approval Process`

`Recheck Conditional Approvals`

Approvals Needed

Status: 2 record(s) found, 1 pages. Click here to re-count records.

`Delete | Save Changes | Cancel Changes  Unlink  Request Reapproval`

	Edit	ID	Step Number ↑	Approval Title	Status
☐	🖉	306	1	Service Responsible Team Approval	Queued
☐	🖉	307	2	CAB Approval for Significant changes	Queued

When the change manager is ready for the approval process to start, she clicks the Launch Approval Process

button, which will set all approvals with the lowest step number to Pending Approval. It will also update the Status of the Change Request to Pending Approval.
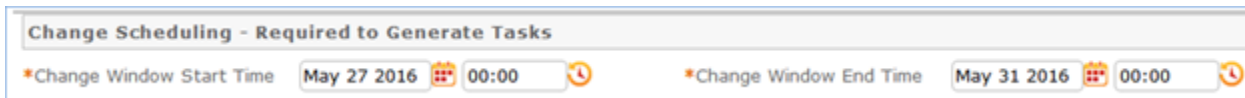
When an approval's status is set to Pending Approval, the approval team and/or Approver are notified by email. Only the person identified as the approver will be permitted to approve or reject the approval. The approval process is controlled by the step numbers. Approvals with the same step number value are launched at the same time, when the approvals with the next lower step number are completed. This is a different model than in the task workflows, in which the sequence field is not functional. With approvals, the step number controls the process and there are no separate sequences or threads in the approval process.

Another way in which approval generation is different from the handling of tasks is that conditional approvals are not created at all unless their condition is met at the time of generating the approval. Since these conditions generally depend on the meta data and fields in the change request, it is important to make any critical fields required to hold a value before getting to the point of generating approvals. The Recheck Conditional Approvals button does allow the technician to recheck the condition, if after generating the first approvals, the meta data changes. But it is best to ensure that any needed meta data is filled out first.

As the first approvals are completed, those with the next step number are set to Pending Approval and their assignees notified. Any approval notes put into the approvals are copied up to the change request in the running All Approval Notes field. Two fields below the approvals table track the Number of Approvals Needed and the Number of Approvals Completed, and once those two numbers are equal and great than or equal to 1, the Status is changed to Approved and the Requester and the Change Management team are notified.

Creating Tasks for the CR
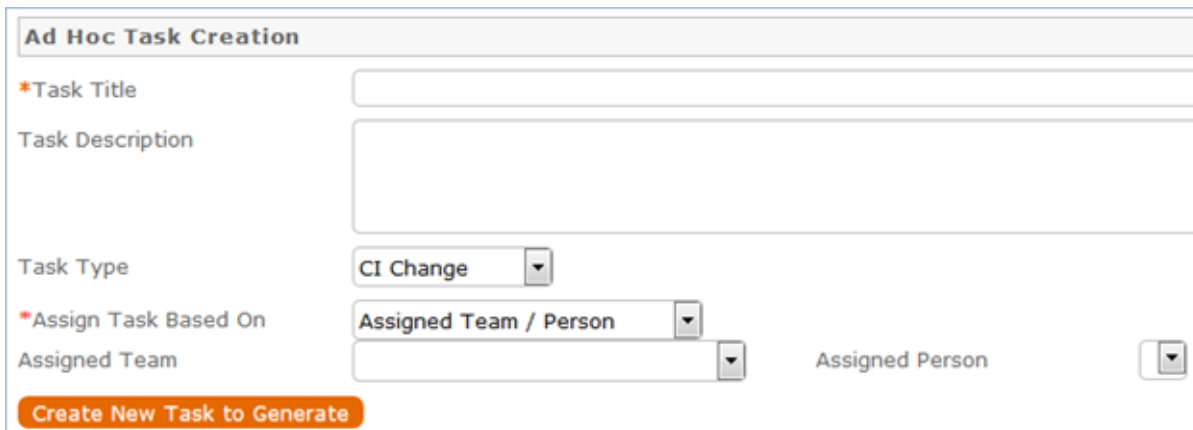
Tasks are created for the CR once any approvals are done. The user must first enter the Change Window Start Time and Change Window End Time, so that the tasks can be scheduled to start on the start date:

Change Scheduling - Required to Generate Tasks			
*Change Window Start Time	May 27 2016 00:00	*Change Window End Time	May 31 2016 00:00

Here the default task template is the one called Upgrade Windows OS. Clicking the Generate Tasks button will generate one task for each of the selected CI's.

Depending on the options selected in the Service, the user may have an option to create more than one ad hoc task. He may also check the box for Create New Task to Generate? to create another task template that can be generated for all the selected CI's. Checking that box will bring up some fields to fill in for the ad hoc task:

Ad Hoc Task Creation

*Task Title	
Task Description	
Task Type	CI Change
*Assign Task Based On	Assigned Team / Person
Assigned Team	Assigned Person

Create New Task to Generate

Clicking the button to Create New Task to Generate will create a new Task Template, save the change request, and reopen it to edit, with the new task selected in the Task to Generate field. From here, you can click the Generate Tasks button again to add the Task to this CR for each CI that is selected.

Launching Tasks

Tasks may not be launched until the approval process is completed. This is to ensure that no work is started without approval. Clicking the Launch Tasks button will set the first tasks (those with no prerequisite or with any prerequisites completed or not needed) to a status of Assigned. It will also set the Status of the Change Request to In Progress, if it is not already in that status.

See the Task Templates and Tasks topics above for more details on how tasks are launched and processed.

Reassigning the Change Request

If a technician needs to reassign the Change Request to a different team or person, he or she simply changes the Assigned Team and/or Assigned Person field and the system will email the new assignee notifying them of the reassignment. Rule: CR - Edit Actions (API Enabled), Action: Assignee Change Notifications.

Completing the Change Request

When all tasks are completed, the Change Manager is notified. Based on whether the tasks were all completed or whether some of them failed, the change manager will then set the Status of the Change Request to Completed and the Closure Category to the appropriate value. If there are tasks whose status is Failed, the closure category will be validated to contain an appropriate value.

Changing the status to Completed will send an email to the Requester informing him that the change is completed. Upon closing a change request, fields on the Progress Notes and Outcomes tab can be used to generate a knowledge article based on the change request.

Setting Review for Knowlegebase to Yes indicates that a knowledge article will be created based on this change request. Once Review for Knowledgebase is Yes, the Knowledge Document Status and Create Knowledge Article fields are visible.The default value of Knowledge Document Status is Pending Review, and once Create Knowledge Article is clicked, the Knowledge Document Status is automatically updated to Created. If it turns out the knowledge article is not necessary, Knowledge Document Status can be changed to Not Needed manually. Clicking Create Knowledge Article opens a new knowledge article record with some information mapped from the change request, such as the Title, Purpose, and Description. Additional required fields must be populated before saving the record. Once the new knowledge article is saved, it is automatically linked within the originating change request. In addition, the knowledge article will automatically include about the source change request.

Working Notes

The Work Notes field is updated from the running working notes for all of the related Tasks.

Related Records and Removing Tasks

During the working process, some tasks may be removed as not needed by the change manager. The action bar above the Tasks table in the change request includes a button to remove one or more tasks. This changes the status of the selected tasks to Not Needed and they move into the Inactive Tasks table shown on the Inactive Tasks tab:

From this area, if it turns out that the task actually is needed, the change manager can select one or more tasks and click the Reassign Task to set their status to Assigned and move them back into the active Tasks table.

Parent/Child Relationship

Previously we used a parent-child relationship of change requests to manage changes to multiple CI's. That possibility is still available but the fields for relating parent and child change requests have been removed from the layout, since that was a much more cumbersome process than the improved process based on generating tasks for each CI within the change request. However, if there is a need to relate change requests in a parent child structure, this logic can be re-utilized by putting the fields back on the layout and enabling the rules that managed the relationships.

Reporting Time Spent

Technician users may easily report the time they spend on handling change requests. There are two fields and an action button: Time Spent, Time Description, and Add Time on the Work Notes tab of the layout. Entering values there will automatically create a new time entry record when the Add Time button is clicked. The time entry will show the work done by the technician and on the current date.

All the time entries for a request can be seen on the Time tab as well. If a technician needs to report time that was spent on a different day or by a different user, he may click the New button on the Time spent table to submit a time entry directly and change the date or "Done by" field. All time entered is totaled in the All Time Spent field, which can be used in reporting or billing.

Note that this same time entry methodology is used in the Incident, Problem, Service Request, and Task tables.

Workflow
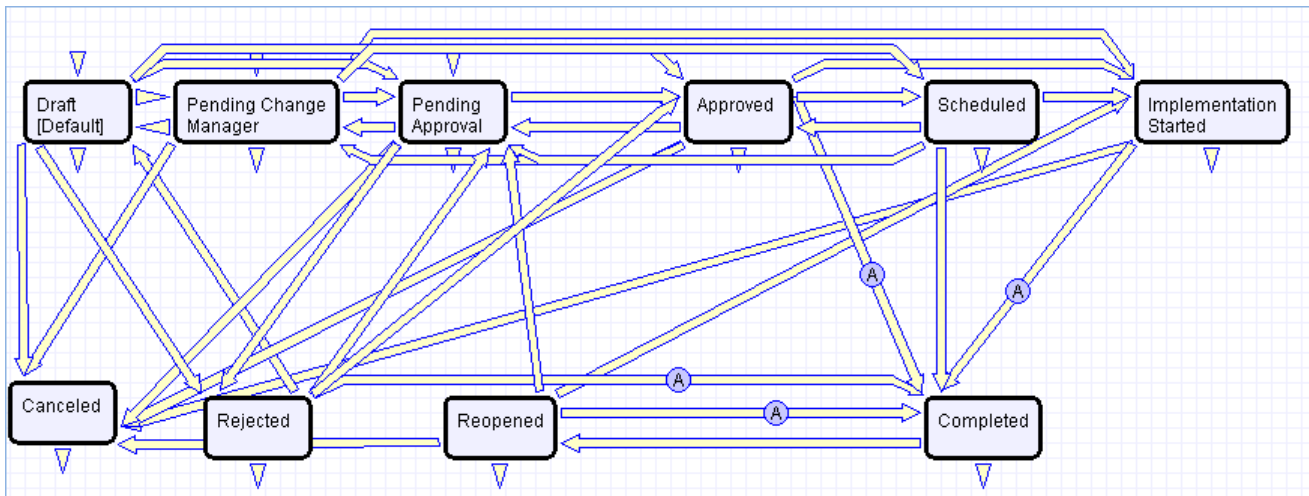
Change Request Form Layout

This section contains an overview and screenshot examples of the information stored in a Change Request in the out-of-the-box system.

Details Tab

A Change Summary must be entered to briefly describe the change, and a Change Category must be selected to categorize the significance of the change. An Assigned Team and Assigned Person can be selected, and anyone who wants to be CC'd on all notifications can be added to the Internal CCs field. If an Assigned Team is chosen but an Assigned Person is not, all team members will receive a notification. If an Assigned Person is chosen, the email will be sent only to that individual.

The submitter fields, Submitter Name, Preferred Notification Method, Phone, Department, Service Level, and Service Catalog in the Submitter Information section default to the person who creates the Change Request, but this can be changed by clicking the lookup icon.

Next, a Business Service must be selected. Depending on the Business Service chosen, certain Services are available to categorize the Change Request further. The Risk for both if the Change is done and not done can be logged for analysis and assessment of the overall risk of the Change Request. Impact indicates the scope of the Change Request's effect, and Urgency can be used to convey how important it is to make the change. Impact and Urgency have options that are available based on the Priority Group for the Service selected. Once Impact and Urgency values are selected, the Set Priority button can be clicked to calculate which Priority to use for the Change Request based on the Impact and Urgency.

A description of the Change is required, and in some cases, a business reasons that justify the change. A backout plan is required, and an attachment or multiple attachments with more details can also be uploaded.

Configuration Items and SLA's

The Configuration Items and SLA's tab shows all of the Configuration Items that will be affected by the change being requested, and allows the user to quickly create these relationships between the Change Request and the CIs. Any relationships created here will also subsequently be displayed when viewing the CI record directly. CI attributes are also shown to facilitate change assessment and authorization, such as the Criticality of each CI.

The Diagram column icon outlined above can be clicked to see a relationship diagram for each CI, showing the other CIs to which it is related:
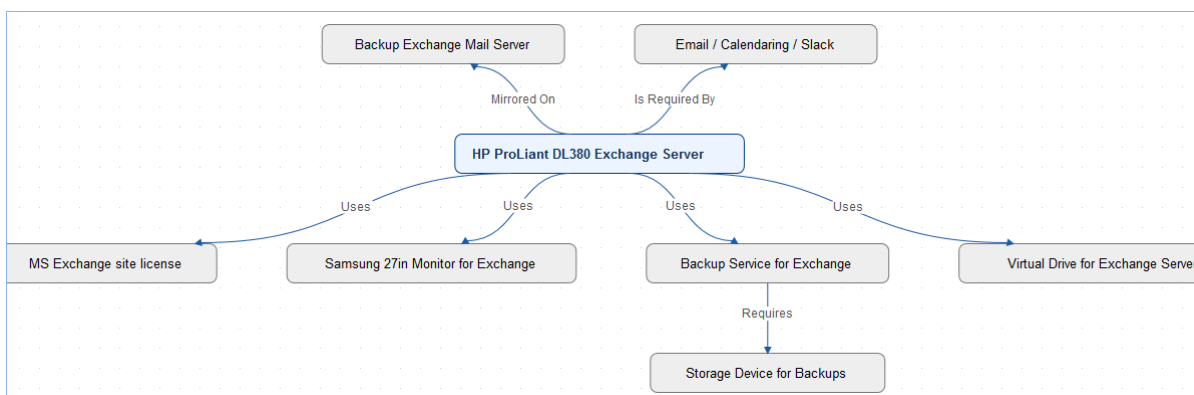
Approvals

Based on the requirements specified for the Service that the Change Request pertains to, approvals may be required. The Approvals tab displays the default Approval Workflow for that Service, and a different can be selected if needed using a lookup. The Approvals are then created and Launched, with the first approval set to a status of Pending Approval. Once all Approvals are completed, the Status of the Change Request is set to Approved.

Schedule and Tasks

The Schedule and Tasks tab displays the Maintenance window for the Service and indicates whether or not there will be downtime. In addition, the Change Request can be tied to a Release record, and information about the Change Window Start Time and Change Window End Time can also be included. Once the change request is ready to be scheduled and has been approved, a button appears to create one or more calendar entries for the change window, as well as for any build or test activities that should be published in the change calendar.

If Tasks are required for the Change Request, they are also shown on this tab with a Default Task Workflow and the option to customize and generate tasks.

Progress Notes and Outcomes

The Progress Notes and Outcomes tab provides a place for working notes to be added. If the Status is changed to a terminal status (i.e. Completed, Canceled, etc.), a Closure Category is also specified for the Change Request.

Related Records

The Related Records tab displays all of the links/relationships to and from the Change Request, including related Problem records, related Incidents, Incidents caused by the Change Request, and related Service Requests.

Ownership

Records in the Change Request table are "owned" by the person whose login matches the Creator Login field.

Reporting and Statistics

This section describes the reports and statistical functions in the Change Requests table.

Forward Schedule of Change and Projected Service Outage

The Forward Schedule of Change represents approved changes and the proposed implementation dates. More generally, it is a list of Change Requests set to occur in the future. Functionally it might include CRs with restricted criteria, such as approved changes only, but by default the Forward Schedule of Change simply shows all Change Request that are not in a status of Closed.

The Forward Schedule of Change is a saved search available from the vertical navigation menu under the Change Request table name. Change Requests are listed in order of Requested Date of Completion. This search is also used in a structured HTML report of the same name. The report groups the CRs by requested date of completion (by week) and shows summary data, including the sum of the Estimated Time to Complete for all records. The report can be configured to be sent in a weekly email to the Change Management team.

Projected Service Outage

The Projected Service Outage represents all approved changes that are set to occur in the future that will result in downtime. Similar to the Forward Schedule of Change, the Projected Service Outage (PSO) is also available from the vertical navigation menu under the Change Request table name, and can be viewed in a structured HTML report. The Change Requests in the PSO are listed in order of the Change Window Start Time.

Both the Forward Schedule of Change and Projected Service Outage can be set up to automatically be emailed to various teams. This also applies to all Charts and Reports, which can be configured to be automatically sent to the defined teams. This allows for the sending of various reports to accommodate pre and post meeting workflows or CAB/eCAB proceedings, such as reports of pending changes, completed changes, and CRs pending post change reviews.

Change Requests can also be easily added to the Change Schedule using the Calendar icon at the top right corner of any record. Users can also subscribe and add the calendar as an internet calendar in Outlook. Users can further refine what they see on their calendars based on the Event Type in the Calendar entries (i.e. Build,

Test, Implementation, etc). A link can also be generated and the same link can be distributed to all users to integrate the schedule into their Outlook Calendars.

TUESDAY	WEDNESDAY	THURSDAY	FRIDAY	SATURDAY	
5 12:00am Switch to new webserver hosting service provider ID:277; 1:00am Upgrade RWC Headquarters connection to use Optical Carrier Transmission (OC-3) ID:278;	6	7 12:00am Provision LastPass Accounts to all employees and collect all paper stored credentials ID:266	8	9	
12 ll employees and collect all paper stored credentials ID:266;	13	14 12:30am	15	16	
19	20	21	22	23	

Accomplished Change

Accomplished Change is the complement to the Forward Schedule of Change and by default shows all closed Change Requests, sorted by Date Closed.

Launch Tasks Button

Description: These actions are run when the button Launch tasks is clicked. If the field Service Approvals Required is set to Yes and there are pending service approvals, the tasks are prevented from launching. If the change request is related to a release and the release Status has not yet advanced to Build in progress or beyond, the tasks are also prevented from launching.

Otherwise, the change request Status is set to Scheduled, a background field that controls visibility of the Launch Tasks button is set to No, the field Launched in all tasks for the workflow is set to Yes, a field controlling visibility of the Create Tasks button is set to No, and the tasks without prerequisites are set to a Status of Assigned.

```
if (Service Approvals Required?='Yes' and (Number of Approvals Needed=0 or Number of Approvals Needed!=Number of Approvals Completed)) {
    Approval process is not completed
} else if (Release ID!=NULL and Release Status>'Build in Progress') {
    Cannot launch tasks until related Release is in Status Build in Progress or greater
} else {
    Set Status to Scheduled
    Set Show Task Launch Button to No
    Set Launched to yes in all tasks
    Set Show Create Tasks Button to No
    Trigger Tasks without prerequisite to Assigned
}
```

Approval Management

There are three tables that control the automated and ad hoc creation of approvals related to various processes within the system. Currently approvals are used within Change Requests, Service Requests, Contracts, and Documents. They can be added to other processes as needed.

The approval processes are managed by the Approval Workflows, Approval Templates, and Approvals tables.

- Approval Workflows Table
- Approval Templates Table
- Approvals Table

Approval Workflows Table

The Approval Workflows table predefines individual workflows for approval processes. A workflow consists of one or more Approval Templates, each of which defines a team or person to be the approver, the order in which the approval will be needed, and some other characteristics of the approval.By default, approval workflows can be used for contracts, change requests and service requests.

The approval process can include a combination of parallel and sequential approval steps.

Use Case

Approval workflow records may be created by members of the Admin, Contract Manager, Change Manager, and Service Manager groups. To create a new approval workflow, click New from the Approval Workflow table action bar or from within the Contract Type or Service record for a change request.

The Related To field must be set to the correct value for the correct linked fields to appear, based on the process for which the approval workflow will be used. The next step is to add approval templates to the workflow by clicking New in the action bar of the Approval Templates related table. For more information on how to design the templates to get a mixture of sequential and parallel approvals, see the Sequence of Approvals.

Required fields are marked by a red asterisk. These include the Status, Workflow Title, and which table the workflow is Used In. If the choice of table in the Used In field is Contracts, additional options allow the user to specify which Contract Types can use this workflow.

When the workflow is selected for use in an approval process, the system will generate required Approval records based on the specified approval templates. Additional ad hoc approvals may be created directly from within the contract or change request if that method is supported, based on the service or contract type setup.

Managing and Reusing Approval Workflows

Approval workflows can be modified even if there are outstanding contracts or services using them. Since the approvals are generated up front as soon as the record moves to Pending Approval, later modifications to the workflow and its templates will only have an impact if there are conditional approvals which are rechecked later in the process.

A workflow can be cloned by using an action button. Clicking Clone Workflow will copy the workflow, and make a copy of each of its approval templates. The cloned Approval Templates will be automatically linked to the newly cloned Workflow record.

Approval Workflow Statuses

The approval workflow Status is either Active or Inactive. An Active workflow appears in the Workflow Title drop-down menu from a Contract or Service record. Workflows with a Status of Inactive are no longer available for use in the approval process; inactive workflows do not appear in the drop-down menu as a possible approval process selection. Admin group members can change a workflow from Active to Inactive if needed.

Ownership

Workflow records are owned by the user who creates them. Specifically, a record is owned by the user whose Login matches the Creator Login field.

Approval Templates Table

The Approval Templates table holds a record for each approval to be created as part of a standard Approval Workflow. It predefines the approval team or person and also controls whether the approval is conditional on some meta data criteria being met or is always generated. Each approval template can only be used by a single approval workflow, i.e., there is a one-to-many relationship between approval workflows and approval templates.

Approval Templates can be created by users in the Admin, Contract Manager, Change Manager, and Service Manager groups. New approval template records are normally added from within an approval workflow record. Creating an approval template from the Approval Workflow table populates the link to the workflow. If an approval template is created directly from the Approval Template table, a linked Workflow Title should be manually selected. Each template contains information about which Workflow uses the approval template, whether the approval is Required or Conditional, and how the approval is assigned. The required fields are Approval Title, Assign Approval Based On, and Step Number.

The Related Records tab displays all other approval templates used in the same workflow. The Step Number and title for the approval are used when converting the template to an Approval record.

Required vs. Conditional Approvals
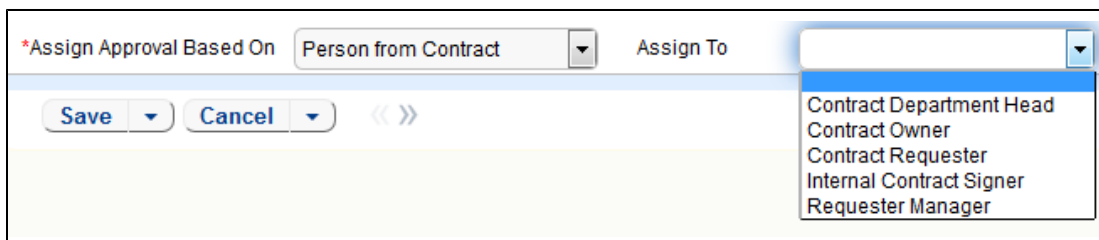
In the Approval Usage field, the user sets whether the approval is Required or Conditional. Both required and conditional approvals are automatically generated when the Create Approvals button is clicked in the contract record or the Generate Approvals from Workflow button is clicked in the change request record.

When Conditional is selected, the Condition field will appear. The user can then input a formula that is evaluated when approvals are generated. This formula can contain any number of conditions that can be linked together by operators such as "or", "and", "contains", etc. Conditional approvals are only generated if the condition is met (typically the Condition is a search criterion based on some field value(s) in the Contract).

Assigning Approvals

Approvals can be assigned to teams or users based on fields in the Approval Template or other variable fields from the parent record such as Contract Owner, Requester Manager, Contract Department Head, or Change Manager, Change Requester, etc. If "Fields in Approval Template" is selected in the Assign Approval Based On field, the Approval Team and Approver fields are visible. The Approver field is filtered to members of the selected Approval Team. If, for an approval template for contracts, "Person from Contract" is selected, the Assign To field appears. The drop-down selection in this field shows a list of user fields from the Contract record, such as Contract Owner, Contract Requester, and Requester Manager.

Note: The selections in the Assign To field are linked from the Replacement Variables table. Refer to the Replacement Variables Table topic for more information.

Similarly, when "Team from Contract" is selected, a team can be chosen from the Assign To drop-down. For approval templates related to change requests, the options for Assign Approval Based On are "Person from Change Request" and "Team from Change Request."

Sequence of Approvals

The sequence of approvals depends on the Step Numbers. Approvals are generated and ordered based on the Step Number order, and are triggered to Pending Approval from the lowest to the highest Step Number. To set up parallel Approvals, give the same Step Number to each Approval Template in a parallel step. All concurrent Approvals (that is, Approvals with the same step number) must be approved in order to trigger the next step.

The automation that controls the triggering and ordering of approvals is managed from the Approval record, not the Approval Template. Fields in the approval record are used to determine if there are concurrent approvals and to define the Lowest Step Number (which may not be 1, if the first step is a conditional approval marked Not Needed).

Automatic Approvals

You can use an approval template simply as a notification rather than an approval. This is done by setting the Auto Approve? field to Yes. When the template is converted into an Approval record, a rule running on the Approval table automatically sets the approval's Status to Approved. If Notify for Auto-Approval is set to Yes, this rule also sends a custom notification message in place of the assignment notification.

Approvals Table

The Approvals table holds all of the approvals that are sent to users and approved or rejected by them. Each record in the table is an individual approval or rejection linked to a parent Change Request, Contract, or Document record.

In addition to being generated automatically based on a workflow and its approval templates, ad hoc approvals may be created by users with the appropriate privileges, either in addition to the predefined approvals, or instead of them.

Use Case for Change Requests and Service Requests

The default status for a new approval record is Queued. There are two main ways in which an approval is created for a change request. The first is by the change manager clicking a button on the Approvals tab:
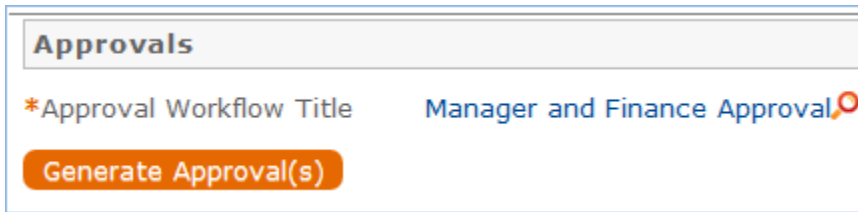
This will generate all of the approvals for that workflow. If an approval template's Approval Usage field has a value of Conditional, then it is not created unless the condition is met. The other method is for a change manager to create an ad hoc approval using the Create Approvals checkbox in the change request on the Approvals tab:
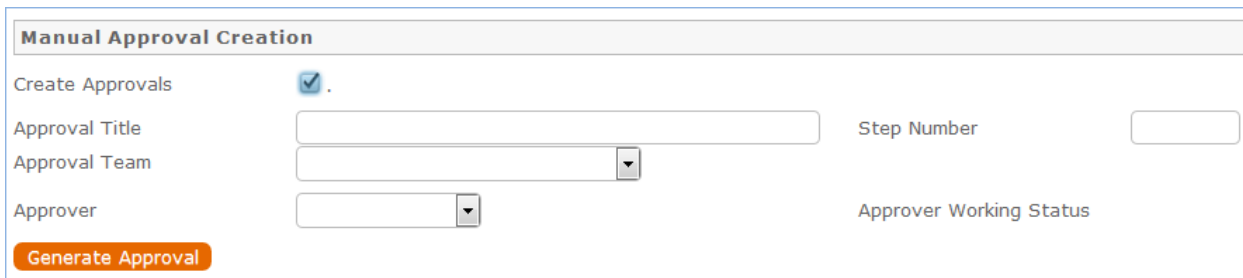
Note that this checkbox is only visible if the service was defined to permit ad hoc approvals and only until the approvals are launched.

The approvers are not notified until the Change Manager launches the approval process, by clicking the Launch Approvals button. At that point, the lowest numbered approval record will be updated to Pending Approval and the Approver or the Approval Team (if the Approver field is blank) will be notified that the approval is due. If there are any concurrent approvals for a particular step, the approval record will be updated with those concurrent approvals.

If an approval step is marked as Auto-Approved, the Approval will be updated to Approved and the Approver or the Approval Team will be notified of the auto-approval, unless the Notify for Auto-Approval field has a No value. This is a method of notifying someone about a change without requiring them to respond.

Approving, Requiring Changes, or Rejecting an Approval

The approver can approve the change by email through a hotlink or click a Require Change or Reject hotlink to edit the Approval record directly and enter some comments and click the appropriate button. If editing the approval record directly, comments will be required if the approval record is rejected or marked as requiring changes. Any individual attempting to click one of the three buttons to approve, reject or require changes will be subject to a validation rule requiring that they be a member of the Approval Team. Whenever the approval notes field is updated, a rule will copy the update into the linked Change Request's All Approval Notes append only field and the field in the approval will be blanked out.

If the approver marks the approval as Requires Change, the linked change request will not be changed, but the Change Management team and Change Manager (if any) will be notified and they will decide whether any changes made require restarting the whole approval process or just carrying on from the rejected approval. If they want to start over, they will click a "Relaunch Approvals" button and that will first set all linked approvals to Queued and then launch the first step again. Otherwise they can set the "requires change" approval back to Pending approval for the process to carry on from there again.

If the approval is marked Approved and there are no concurrent approvals, the next approval record in the sequence will be updated to Pending Approval and the Approver or Approval Team will be notified. If there are concurrent approvals, all concurrent approvals must be completed prior to the next approval step being updated to Pending Approval. The approver and the date approved will be updated to reflect whoever clicked the Approve button.

If the approver marks the approval as Permanently Rejected, the linked change request will be updated to a status of Rejected and the Requester and Change Management Team notified. The user will be required to resubmit the change request if still required, with appropriate changes. All other approvals in the Change Request in a status of Queued or Pending Approval will be updated to Not Needed. If an approval is updated from a status of Pending Approval to Not Needed as a result of the above action, the Approver or Approval Team will be notified that their approval is no longer necessary.

Create Approvals

The process begins when the Create Approvals button is clicked in the contract record. Contract approval records are created from Approval Templates using a conversion action or manually on an ad hoc basis. For information on starting the approval process for a contract, refer to Handling Approvals.

Approval Records

Each approval record stores the parent Contract ID, Approval Team, and Approver, the user who submitted the approval. The Date Approved/Rejected field captures and displays the timestamp of the approval, shown on the History tab. Notes about the approval or rejection are entered into the Approval Notes field. When the record is saved, the notes are appended to the All Contract Approval Notes field and are visible from any approval linked to that Contract. Additionally, the notes are appended to the Approval Notes field in the parent contract record, located on the Approvals tab.

All Notes by Approvers	
Approval Notes	[Agiloft System Mar 31 2015 10:56]
	This contract needs significant revision.

Only Approvers and members of the Approval Team can update approval records. Users updating the Status to Permanently Rejected or Requires Change must enter Approval Notes.

Use Case for Documents

To begin the approval process for documents, select Yes for Requires Approval on the Progress tab of a Document record. Select one or more Reviewer(s) who will be assigned to review and approve the document. Click Submit for Approval to generate the approval records.

All Document approvals are created with an Approval Status of Pending Approval. Document approvals are parallel, i.e. all document-related approval records are created and a Step Number of "1" through conversion. Notes added to document approvals are appended to the Approval Notes field in the parent document. For more information on approvals for documents, see: Documents Table.

Ownership

Approval records and Approval Template records are owned by the user who creates them. Specifically, a record is owned by the user whose Login matches the Creator Login field.

Task Management

There is a set of tables used to manage the automated creation of tasks that may appear in service requests, change requests, projects, releases, and other places within the system. These tables, which are described in the following sections, include:

- Task Workflows Table
- Task Templates Table
- Task Steps Table
- Tasks Table
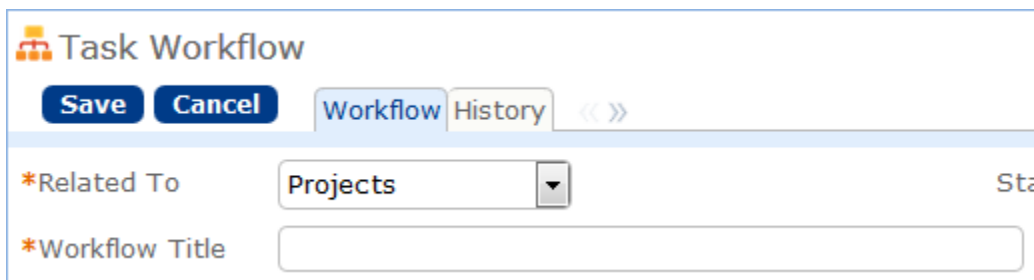
Task Workflows Table

Task Workflows are used to create a set of task templates that may be organized to trigger all at once or in a specified sequence. They can be used in services such as Change Requests or Services Requests, and they are also used in Project Types. They essentially predefine the tasks that should be done for a particular request type. They are used in conjunction with the Task Generation Method of Predefined Task Workflow.

Task workflows can be created by members of the Project Manager, Change Manager, Service Manager, and Admin groups. All except admin users may only edit the workflows related to their primary tables, so change managers can edit workflows related to change requests, while project managers can edit workflows related to project types.

Creation of Records

A task workflow can be created directly in the Task Workflows table or it can be launched from within a service for a change request or service request or from a project type record, using an action button. The advantage of doing it from a service record or project type record is that the Related to field will be populated automatically with the correct table:

Either way, the process of setting up a new task workflow involves naming it with a unique name, ensuring it is related to the correct table, adding a description, and then creating the task templates. Clicking the Create New Task Template button saves the workflow and then brings up the new task template screen. See Task Templates Table for more information on how to create and fill out the task template form.

Once the first template is created, if the Enable Prerequisite Tasks is set to Yes in the workflow, then additional templates may select the earlier ones as prerequisites.

Prerequisite Task Handling

Within a single task workflow, there may be multiple threads of dependent tasks with different prerequisites, and these threads may proceed independently. It is possible to launch two separate threads and bring them back into alignment by making their last tasks prerequisites of the next task, for instance.

Task templates can be conditional, and if their condition is not met, then they will be marked as not needed when they would otherwise be assigned. When they are marked as not needed, their dependent tasks will be assigned. This is an important point: tasks that are dependent on conditional tasks are not prevented if the condition is not met. If they should be prevented, then they should also be made conditional based on the same condition as their prerequisite task.

Note that when tasks are launched for a project or a request using a Launch Tasks button, the Status of tasks will be changed to Assigned based on the following logic:

- The task status was Queued or Conditional
- The task has no prerequisites and is either not conditional, or its condition is met
- The task has some prerequisites but they are set to a status of Not Needed or Completed – i.e. they are conditional and are not needed, or they have been "removed" (marked as not needed) by a technician before the tasks are "launched"

Below is an example of a task workflow for employee termination:

Task Workflow Title Employee Termination ▾   **Refresh Workflow List**

Status: 5 record(s) found, 1 pages. Click here to re-count records.

Edit	ID	Sequence ↑	Task Title ✎	Assigned Team ✎	Assigned Person ✎	Assign To	Prerequisite Titles
☐ ✎	62	1	Turn off access to network and vpn	System Administration Team			
☐ ✎	61	1	Hold exit interview	HR Team			
☐ ✎	160	2	Send an email to other employees notifying them of the termination	HR Team			Hold exit interview
☐ ✎	64	2	Reassign all support and other tasks	2nd Level Support Team			Hold exit interview
☐ ✎	63	3	Edit telephone extension to forward to manager	System Administration Team			Hold exit interview Reassign all support and other tasks

Here there are two tasks that will be assigned right away (ID62 and ID61 - those with no prerequisites). The tasks with a sequence value of 2 will be assigned as soon as the Hold Exit interview is completed, so they might get assigned before ID62 is completed. The final task will be assigned as soon as the exit interview is completed and the support and tasks are reassigned. So it is conceivable that ID62, ID160 and ID63 are all in progress at the same time and any may be completed first.

The ability to choose exactly which tasks must be finished before a task is assigned gives a very flexible model for complex task workflows. The sequence number gives a general impression of the order of tasks, but it is not definitive and does not control any of the processing. It allows the tasks to be sorted more or less in order of assignment.

Cloning a Task Workflow

Workflows can also be cloned using the Clone Workflow button. The effect of cloning a workflow is to create a new workflow with the workflow title changed to prefix it with information that is was cloned, i.e. Cloned on 05/27/2016 02:58 - General IT Project. The task templates are also cloned and linked to the new workflow.

Note that task templates may only be linked to one workflow and are created independently for each workflow, since they may have a sequence and order within that workflow that would be different within a different workflow.

Task Templates Table

This table holds records that serve as templates for automatically generated tasks. Each Template record specifies a Task related to one of the other tables. Task Templates have a Related to field, just as tasks do, and they may be linked to a specific service, task workflow, or project type. They define the method of assignment, the expected number of working hours to complete, and any prerequisite tasks.

Task Templates may be combined within a Task Workflow, or they may exist independently or a workflow. They can be combined to form a set of "User Selected Tasks" linked to a service or project type. Or a single task template may be linked to a service for a change request.

Template Details Tab

The user entered fields are all on the Template Details tab:

The title may not contain any commas. Task templates may be related to Service Requests, Change Requests, and Projects. The Task Usage may be Default or Conditional. If conditional, then a saved search condition based on meta data in the record where the task is generated should be defined. At the point where the task would be assigned, the condition is checked to see if it is met or not and the task's status is changed accordingly. The Number of Working Hours to Due date is used to set the task's Date Due once it is set to a status of Assigned, by adding the defined number of working hours (for the assigned team) to the current time.

The Assign Task Based On allows you to define a task template whose resulting task will be assigned to a person defined in the main record. The default of Assigned Team / Person lets you "hard code" the assigned team or person in the task template. A different value allows you to choose a variable that has been preconfigured in the Replacement Variables table. For instance, for a task related to a service request you will see these options:

The choices in the Assign to field are set in the Replacement Variables table and can be edited there, or new fields added.

Note that if you set a task to be assigned in this way, but there is no value in the selected field – for instance, there is no submitter manager defined in the service request-- then the task will not be properly assigned. As a backup, such tasks are assigned to the 1st level Support Team.

The Choose a Task to Add field is used to select a task for the same workflow to be a prerequisite task to this one. The Prerequisite Task to Remove is used to remove a task that was previously defined as a prerequisite. These fields are used in conjunction with the action buttons to their right.

Note that prerequisite task setup is only visible and enabled if the task template is linked to a Task Workflow and if the linked field Workflow Enable Task Prerequisites=Yes. Otherwise these fields will not be visible.

At the bottom of the tab is the option to set up a task checklist. If Yes is selected, then you can add individual steps and define the order the checkboxes should use:

To add steps, first check the Create New Checklist Item? Box, then fill out a name and the step number, then click the **Create Task Step** button. This is a handy way to provide a list of troubleshooting steps or a list of things to be done within a single task.

Related Information Tab

This tab may show the record that contains this task template. If the template is part of a task workflow, the workflow will be listed. If it is linked to a particular service as a "User Selected Task", then the service name will be linked in:

Use Case

Task Templates are related to either the Project, Change Request, or Service Request tables using the "Related to" field at the top of the form. Whenever a Task Template is used to generate a Task, a conversion is done from the Task Template table to the Task table, mapping important information in the Task Template to the Task record that is generated. This Task record is linked to the Service Request, Project, or other table that spawned it, as well as being linked back to the source task template.

Templates can be made inactive when they should no longer be used, and that will prevent them from being

generated as tasks. When the Status is changed to Inactive, an edit rule called Edit: Other edit actions - unlink inactive task template, handle cloned templates will blank out any link to a task workflow, project type, or service will be eliminated so it will no longer appear as available in those services or new projects. Another rule prevents task titles from having commas in them, and if a comma is stripped out, provides a popup message alerting the creator that it has been stripped.

Automation and Workflow

The automation on the Task Template table is almost all designed to handle the conversion into tasks when triggered from elsewhere, rather than to manage a process within the Task Template itself. It is described in the appendix. In general, there are "flag" fields in the task template that get set to Yes to trigger conversion for each of the different main record types, and linked fields to the source records so that the new tasks that are triggered can be linked to the correct service request, project, or change request. In addition, we keep track of all service requests and projects for which a task template has been generated so that if a user tries to generate the same task for the same parent record, the system prevents it. Since change request tasks must be generated multiple times, one for each CI, we cannot block this in the same way.

Cloning Task Templates Along with a Task Workflow

Task workflows may be cloned and then modified, and when a task workflow is cloned, its task templates are also cloned and linked to the new workflow. Cloning reproduces all the elements of the original task template except any task steps. If task steps were involved in the task template, they will need to be recreated in the cloned template.

Task Steps Table

Task steps are used to define a multi-step process within a single task rather than creating several tasks. This makes sense especially when the same person is performing all the steps. Individual records are linked to a specific task template and can only be used in that template. They are very simple records with just a title and a sort order. They appear within task records as a set of checkboxes that are ordered based on the sort order:

Task steps are always created from within a task template, as shown above in the task template section. They should not be created directly, since without being linked to a task template, they will never appear anywhere. There is no automation on this table.

Tasks Table

Like Services, the Tasks table is a global table that holds records that are often linked to and appear within the records in other tables. The Tasks table holds individual tasks. While it may be used for tasks of many different kinds, the system is currently set up to allow tasks to be related to Service Requests, Change Requests, and Projects, as well as to Configuration Items, or to be completely independent of other tables. It is possible to modify the setup to relate tasks to records in any table and to show embedded tasks in any other table.

Fields related to task generation and an embedded Tasks table (labeled Tasks) are shown in service requests and change requests for services that include tasks. Tasks and the fields for generating them are also shown within Projects and Configuration Items. Projects allow the same methods of task generation as service requests, while Configuration Items allow single tasks to be created and linked to the CI, and they also show all tasks that have been created from other records (change requests or service requests) that relate to that CI. Within the task table, the field Related to indicates what type of record the task is associated with.

If you need to use tasks within any additional tables, you can modify this field and add that table to the choice list. Then create a linked set of fields from that other table, and make these fields visibility dependent on the value in the Related to field just as we have done for the linked Service Request and Change Request fields. Then create a related table in the other table pointing to the Tasks table, and you will be able to generate tasks from there. This advanced functionality is best done after attending a training class or with our professional consulting team's assistance.

Task Details Tab

The main task screen has fields for assignment, the Task Type, its Status, its Date Due, and so on. It also includes an area for managing the related configuration item, if any, and a place to add working notes and to display all the running working notes that have been added.

When working on a CI-related task, the technician can click one of the buttons shown above to update the Operational Status of the CI to reflect that he has taken it offline or brought it back online. Some tasks may have Task Steps defined. These steps are set up in the task templates. Task steps will appear as checkboxes above the Working Notes field. There are no default rules enforcing that all checkboxes are checked before completing a task, but such a rule could be easily added.

Related Tasks Tab

The Related Tasks tab shows the prerequisite tasks, if any, and allows the current task to be related to prerequisite tasks within in the same parent record. It also shows any dependent tasks - those for which this task is a prerequisite.

In the above example, we see a task that has two prerequisites already defined. The Trigger Condition defines whether this task will be set to Assigned only when both prerequisites are done, or as soon as any of them are done. Additional tasks associated with the same Service Request can be added to the prerequisites by selecting the task and clicking the Add to Prerequisites button. Either of these two tasks could be removed by selecting it in the Remove Task from Prerequisites field and clicking the Remove from Prerequisites button.

The default Status for a task is Queued when it is created. Typically a rule will set the Task to Assigned when the tasks for a record are launched (see below for more details).

Related Info Tab

This tab shows details about the record linked to the task, which will typically be either a project, service request, or change request. It provides hyperlinks to get to the source request to see more information.

Other Tabs

The Time tab allows time to be entered and shows all time entered for the task. Note that any time entered for the task will also roll up into the request to which the task is linked. The Emails and History tabs hold the standard fields.

Using the Tasks Table

This section describes how to use the Tasks table.

Task Creation from a Template

Tasks are most often created when a user clicks a button to Generate Tasks from the record in which the tasks will be done, i.e. from within a Project, Service Request, or Change Request. Such tasks are generated from Task Templates that have been created previously and defined to be used for the particular project or service type. When generated from a template, they will be auto-assigned to the appropriate team or person based on the task template record.

Note that currently, users are prevented from creating tasks whose task title has a comma, since this breaks some of the automation for prerequisite tasks. If a task is created with a comma, the comma is stripped out. Commas are also prevented in task templates for the same reason.

Ad Hoc/Manual Task Creation

Tasks can also be created manually outside of any other record or by clicking a button to create an ad hoc task from within one of the request records. In this case, the user will choose the Assigned Team and/or Assigned Person and may also set a Date Due, select prerequisite tasks, and so on.

Creation Automation

When a task is created manually, if it is related to a Project whose status is Completed or Cancelled, the user is prevented from saving the task. If the Date Due is in the past when the task is created, the user is warned but allowed to correct or save the task. These actions are done by the rule: Create: All create validations.

Next a rule called Create: All Creation Actions runs, and it performs several actions based on the record the task is related to. If the task template usage is Conditional, then the Status of the task is set to Conditional. Otherwise the default status is Queued. If the task was assigned to an individual from the related record (i.e. the Change Manager or Project Manager), then the Assigned Team is set to that person's Primary Team. If no

one was assigned at all due to some failure of the template, then the Assigned Team is set to the 1st Level Support Team.

If the task...

- Is created in a Status of Assigned, then if the Assigned Person is not the creator and there is an assigned person, that person is emailed, otherwise the assigned team is emailed.
- Is for a change request and was generated as a single task, its sequence value is set to 1.
- Source is a task template that had prerequisite templates, then the corresponding tasks are set to be prerequisites. The prerequisites are then sorted and the one with the highest sequence value is set in another linked field called Highest Sequence.

A rule called Create/Edit: Update Sequence based on Highest Sequence than runs to set the new task's sequence value to the highest sequence plus 1.

Note that the Sequence field is purely informational. No automation is triggered based on the sequence, but it is there to provide a general idea of the order in which tasks will be triggered and completed.

Processing a Task

The person assigned to a task can add working notes to it, refer to the linked Service Request, Project, or Change Request from it, and ultimately complete the task. There are several default statuses: Queued, Conditional, Assigned, Completed, Not Needed, Failed, and Waiting for Others.

When the person has completed the task, he may change the Status to Completed and save. If there is no value in the Date Done field, the system will put the current date/time into the Date Done field. He may alternatively choose to enter a time in that field directly. The system allows the user to override the value in that field. While working on the task, the user may enter any time spent on the task in the Time Spent and Time Description fields, then click "Add Time" to convert them to Time Entries.

When working on a CI, the Start Clock and End Clock buttons can be used to set the Actual Start Time and Actual End Time as needed. The user may also manually put values in these fields. When a task record is saved the rule called Edit: All Edit Actions (API enabled) is run. If the Status has changed to Completed, Failed, or Not Needed, the system updates the Dependent tasks by refreshing their Number of prerequisite tasks counts. If this was the last prerequisite, then that will trigger the next rule to assign the dependent tasks.

The rule also sets the Assigned Person, if it is blank, to the person who completed the task. If the task is for a project, the project manager is notified of the task completion. If the working notes field was updated, the text is copied into the Running Notes field and blanked out. And if the task is related to a change request, the notes are also updated into the Change Request's running working notes.

If the status has just changed to Assigned (by launching the tasks from the main record or by the prerequisite tasks having been completed), the Date Due is set based on specified criteria. If the task was an auto-completing task, then it is marked as Completed and the assignee notified. Otherwise, the assigned person or assigned team is notified that the task is now assigned.

When a prerequisite task is completed, its dependent tasks are updated, and if all prerequisite tasks are now completed, then the rule called Edit: Assign tasks when number of completed prerequisites meets criteria (API) runs. This rule checks if the task was conditional, and if so, checks the condition to see if it is met. If the task is not conditional or its condition is met, then it sets the Status to Assigned. Otherwise, it sets the status to Not Needed. When all tasks for a particular record are completed or marked as failed or not needed, the person / team assigned to the main request is notified that all tasks are done.

Measuring Time for a Task

In addition to time that is manually entered, the system tracks two kind of elapsed time. The Working Hours to Complete field is set to the difference between the Date Created and Date Done, excluding the non-working hours of the assigned team and also excluding the time during which the Status was Queued, Conditional, or Not Needed.

The Actual Working Hours field measures the time between the Actual Start Time and Actual End Time, excluding the non-working hours of the assigned team. These fields can be used in reports to see the average amounts of time tasks of specific types are taking. They can also be compared against the Template Number of Hours to Due Date value, which sets the expected working hours that should be needed for the task.

Automation and Workflow

There is a simple workflow for tasks that currently executes no actions:
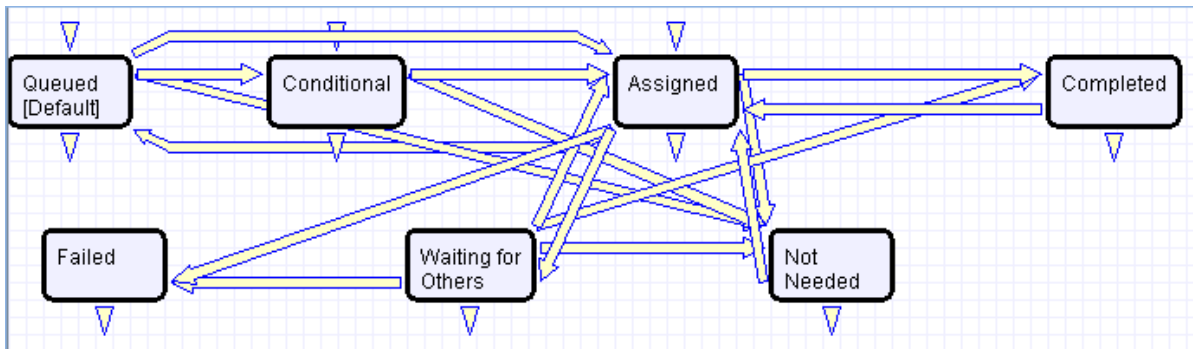
Time Based rules

There are two time based rules that are set up but not running:

- TB: (DISABLED) Notify of upcoming task. It will notify the assigned person or team when the due date is one day away, once it is turned on. Currently it is disabled. There is a radio button at the bottom of the General tab in the table settings if this rule is desired. The schedule may need to be changed from every 10 years to something more useful.
- TB: (DISABLED) Set alert color to red if overdue. This sets the alert color to red when the date due has passed, so that views that use row coloring can show to users that the task is overdue.

Impact, Urgency, and Priority Management

There are five tables that manage the urgency, impact and priority values that are shown within a service request, incident, change request, or problem record. This section describes how these tables work.

Priority Groups Table

Each service in the Service Portfolio table is linked to a priority group. Here is the setup screen for the service on the SLA / OLAs / Priorities tab:
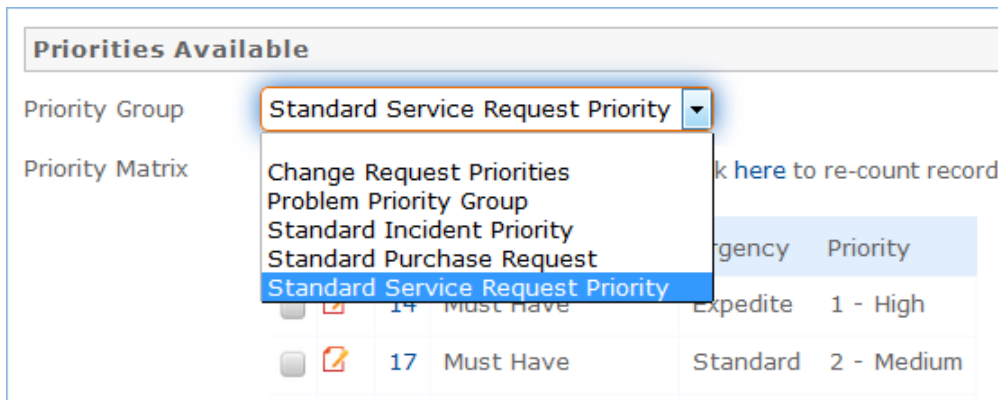
By default there are five priority groups that may be used, but you can create additional ones as needed. They are applied based on the request type and each priority group can have its own set of urgencies, impacts, and priority matrix of values that result from all combinations of urgency and impact within a record. For example, here is the priority matrix for standard service requests:

Priority Group

Save ▾ Cancel ▾ | Details | Related Info | History | « »

| ID | 2 | Status | ● Ac |

Priority Group Details

*Request Type   Service Request ▼

*Group Name   Standard Service Request Priority

Priority Matrix

Status: 6 record(s) found, 1 pages. Click here to re-count records.

New | Mass Edit | Delete | 🔍 Actions ▾ | Views ▾ | Search ▾ | ✉ ▾ |

☐	Edit	ID	Impact	Urgency	Priority
☐	✎	14	Must Have	Expedite	1 - High
☐	✎	17	Must Have	Standard	2 - Medium
☐	✎	15	Should Have	Expedite	2 - Medium
☐	✎	18	Should Have	Standard	3 - Low
☐	✎	16	Whenever Possible	Expedite	3 - Low
☐	✎	19	Whenever Possible	Standard	3 - Low

| Number of Priorities | 6 | Expected Number of Priorities | 6 |

All of the wording in these fields is defined for the specific priority group, so it is possible to have completely different values in different services, as needed.

Note: In order to set up a new priority group, the priority group must first just be created, named, and saved. Then, either the existing impact and urgency records need to be updated to add that priority group to the available groups, or new impact and urgency records can be created and linked to the new group. That will make those values available to create the priority matrix.

Impacts Table

The Impacts table contains all possible Impact values that can be selected in the request tables. Each impact can be made available for one or more priority groups. Within a request, the available impacts are filtered based on the priority group of the Service for the request being contained in the Impact record's Available for Priority Groups field.

The Order field defines the order in which the impact will be listed in the dropdown list. The out-of-the-box impacts available are shown below:

Impacts

Status: 13 record(s) found, 1 pages. Click here to re-count records.

New | Mass Edit | Delete | Actions ▾   Views ▾ | Search ▾ | ✉ ▾ | 🖨 ▾

	Edit	ID ↓	Impact	Order	Status	Available for Priority Groups
☐	✎	15	11 plus incidents	1	Active	Problem Priority Group
☐	✎	14	5-10 Incidents	2	Active	Problem Priority Group
☐	✎	13	1-4 Incidents	3	Active	Problem Priority Group
☐	✎	12	One or More Individuals Affected	4	Active	Change Request Priorities
☐	✎	11	Department Affected	3	Active	Change Request Priorities
☐	✎	10	Entire Building Infrastructure	2	Active	Change Request Priorities
☐	✎	9	Company-Wide Infrastructure	1	Active	Change Request Priorities
☐	✎	7	Whenever Possible	3	Active	Standard Purchase Request Standard Service Request Priority
☐	✎	6	Should Have	2	Active	Standard Purchase Request Standard Service Request Priority
☐	✎	5	Must Have	1	Active	Standard Purchase Request Standard Service Request Priority
☐	✎	4	Individual	3	Active	Standard Incident Priority
☐	✎	3	Department	2	Active	Standard Incident Priority
☐	✎	2	Company	1	Active	Standard Incident Priority

Urgencies Table

The Urgencies table contains a record for each possible urgency value that can be selected within a request record. Urgencies available for selection with a request are filtered based on the Service Priority Group being contained in the Urgency record's Available for Priority Group field. Below is a list of the default urgencies that have been defined:

Urgencies
Status: 7 record(s) found, 1 pages. Click here to re-count records.

Edit	ID ↓	Urgency	Order	Status	Available for Priority Groups
	8	Emergency	2	Active	Change Request Priorities Problem Priority Group
	7	Standard	2	Active	Standard Purchase Request Standard Service Request Priority Change Request Priorities...
	6	Expedite	1	Active	Standard Purchase Request Standard Service Request Priority
	4	Low	4	Active	Standard Incident Priority
	3	Medium	3	Active	Standard Incident Priority
	2	High	2	Active	Standard Incident Priority
	1	Critical	1	Active	

Priority Values Table

The priority values table holds records for all the possible priority values that will be used in the priority matrix, to make it quicker to set up new priority matrices. The default values are:

Priorities Table

The Priorities Table is where it is all put together to define the matrix and how priorities will be assigned based on the combination of impact and urgency. Priority records are best created while editing the Priority Group record from the related table of priorities shown there. Here is an example of a change request priority record:

To set up the matrix, create a Priority record and choose an impact and an urgency and decide what priority you want to be the result of that combination. Then create the other records. When creating a Priority record, the impacts and urgencies available for selection are filtered based on the priority group, which is why they need to be set up first. To change the logic that leads to a certain priority, the Impact and/or Urgency fields

within the priority record can be modified.

Service Catalog and Portfolio Management

This section describes the out-of-the-box structure for the Service Catalog and service management portion of the Agiloftsystem. The primary tables involved in managing services are:

Table name	Description
Business Services	This table holds the high level business services for which specific service offerings will be made available to users based on specific SLAs. The business services are linked to configuration items to provide a diagram in the configuration items table of the relevant components and services on which the business service depends. An example of a Business Service is "Network Services" or "HR Management". For more information, see Business Services.
Service Portfolio	This table holds all of the specific service offerings that are being planned or are in progress – the Service Pipeline – as well as those that are Active or being reviewed for renewal – the Service Catalog – and those that are retired. Each individual service is linked to a Business Service. Under the Network Services Business Service, for example, are services for Can't Access the Internet, Can't Access Shared Drive, Configure wireless access, Report slow network or internet connection, and so on. Each service is used in a specific request type, such as Service Request, Incident, or Change Request. For more information, see Service Portfolio Services.
Service Plans	This table is used to capture information for Service Improvement Plans and Service Quality Plans in support of continual service improvement. These plans can be linked to customer surveys and to the CSI Register.
CSI Register	This table is used to manage a CSI Register and collect information there that will be linked to Service improvement plans.
Service Level Agreements and Underpinning Agreements	These tables will be discussed in the Service Level Management section, though they have direct relationships to the Service Portolio and Catalog.

Business Services

This table holds the primary business services under which specific service offerings are organized. It includes fields for collecting data about the supporting services, the business use case, and the SLA customer groups for whom the service is available, which is automatically set based on the SLAs that have been linked to the business service. By default the service catalog consists of a two-level hierarchy with a Business Service being the parent and Services being the children.

There are additional subsets within this hierarchy based on the Request Type resulting from the service – Service Request, Incident, or Change Request – and also, within Service Requests, based on whether the service involves a purchase, defined by the field "Involves Item Purchase- Yes/No" in the service table. Further tagging or keywords could be used to segment the catalog in other ways as needed. Each business service includes a field to hold an icon that will represent it in the displayed service catalog.

Sample Display in Service Catalog

The out-of-the-box system has several sample business services, some of which are shown below in the main service catalog view:

Icon	Business Service ↑	Description
	Account Management	This Service pertains to controlling access to network resources or centralized software. This can mean updating permissions, or creating new accounts.
	Company Website	This service includes management of the company websites and web pages and all content. It includes services for requesting changes to the website, correcting errors, and managing versions.
	Conferencing/Presentation	This Service encompasses Conference Setup, Web sharing, Projectors and more.
	Desktop Hardware	This Service covers the requisition, upkeep, and disposal of standard computer systems, desktops and laptops to end users. It also includes standard additions to the computer, such as OS, Anti-Virus and miscellaneous software utilities in use by the organization.
	Desktop Software	This service includes maintenance and upgrades of all standard desktop software applications and management of their licenses.
	E-mail / Calendaring / IM	This Service provides email and calendaring applications and infrastructure to enable full internal and external communication capabilities. It includes management of both personal and group calendars, email aliases, mailboxes and archiving. It also covers usage of the company Slack account for instant messaging....
	Enterprise Apps	This Service contains options for mission-critical business software used by the organization.
	Facilities Maintenance and Security	This category includes services and problems related to HVAC, Lighting, furniture, key cards, and other facility-related issues.
	HR Management	This Service is used primarily by HR and hiring managers to set up new employees, update their information, or request an employee separation task.
	IT Infrastructure Change	This is a technical service only available to IT staff.
	IT Service Issues	Sometimes, a system will not respond the way one would expect, or not respond at all. This option allows you to submit an outage report to let us know if something is not behaving correctly. This service is used when none of the other main categories seem to apply.
	Network Services	This Service covers the network itself, both wired and wireless, and also includes file sharing and backup/restore services.

Business Services Layout

The Business Service table consists of the following screens:

Details Tab

This tab holds the main details about the purpose of the business service, its general availability and maintenance hours, the request types for which it is available, and so on.

Note that the Availability and Maintenance schedules are defined using the working hours of a team and the fields belong are actually links to the Teams table, with the team name describing the time frame being used. If the time frame needs to be changed, the team can simply be edited, or new teams may be created with the appropriate days and hours.

In addition to Availability and Maintenance windows for the business services, support hours are defined for services with the SLA record, and those support hours, also defined as teams, are used when setting SLA targets. See Service Level Management for more details.

The Business Service table is used to on-board new business services as well as to define and display the active ones to users. The visibility of a business service to customers and non-service professionals is controlled by the Status field, which has the following values:
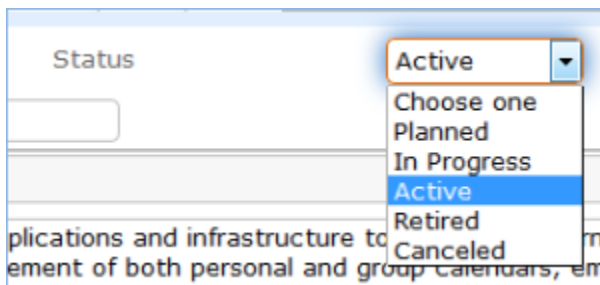
General users can see only business services with an Active status, while the service management team can view all records and can use the other statuses while working on development of a new business service. This tab also includes the list of Services that are available related to this business service, as well as a count of how many services for each request type are active.

When selecting a business service within a particular request type, only those that have at least one service for that request type will be shown. So for instance, the business service shown above would not be visible when creating a change request. The Icon field is used to define the icon to be displayed to users in the catalog.

SLAs / OLAs Tab

This tab displays the SLAs and OLAs that have been linked to this business service. Note that a service may have multiple SLAs with differing levels of service and different service targets linked to it.

Customers With Access to the Business Service

Access to the business service is controlled through the SLAs and to the SLA Customer Groups with access to the SLAs. This information is summarized in the Available for SLA Customer Groups field, which is automatically updated with all the SLA customer groups that are contained within the linked SLAs. This can be used to get a full list of which customers have access to this particular business service.

In order to view which individual users have access, simply click on a link in the Available for SLA Customer Groups field. This will bring up a view of that SLA Customer group record. From there, click on the Related Info tab to see a list of all customers for that SLA, and hence, this business service.

Ongoing Notes Tab

This tab holds an append only field in which to put ongoing notes about the business service. The note is time and date stamped with the name of the user who last updated it, for a running history of updates.

Configuration Item Tab

Each business service is also created as a configuration item so that its relationships to components and other CIs can be setup and displayed where needed. This information appears on the Configuration Item tab. The list of the direct CI relationships to this service are displayed as shown above for the Email /Slack service.

Clicking on the Associated CI ID hyperlink will take the user to a view of the linked configuration item, from where a CI diagram can be displayed showing the relationship of this service to other configuration items, down to as many levels as have been configured.

Email and History Tabs

These are the standard tabs that show all email communications from the business service record and the history of all changes made to the record.

Automation

Two rules run each night that refresh the values in the Available for SLA Customer Groups field by matching on all the linked SLAs and the groups within those SLAs. This handles a situation in which a new group is added to an existing SLA, such as a Corporate SLA or a new SLA becomes active and is linked to this service. These rules could be set to run more often in a rapidly changing environment.

Service Portfolio Services

The Service Portfolio table holds all of the individual service offerings related to a business service - the individual records in this table are called Services.

It is the main source for the detailed ITIL Service Catalog. It holds a record for each service that may be offered to end users or technicians through the service catalog. All services offered by the organization to end users should be managed here and this table's workflow is used to manage the on-boarding process for a new service. Each service is offered through a different request type, such as a Service Request, Incident, or Change Request. Each service is therefore associated with one of these three tables, and this association is defined in the Services field "Request type."

Presenting Services to Customers

There are several ways the system can be configured to enable users to create a request for a particular service. Whichever method is used, users will only see the active or in review business services and service offerings for which their SLA customer group has an active SLA.

The system comes out of the box with Corporate SLAs that cover all users and that are linked to most of the default services, but each service may have specific SLAs available to only one set of users. An example of such services in the default system are the Quickbooks services, which are available only to the Finance Team SLA group.

Whenever a customer user interacts with the services or the service catalog, that user will only have access to the services that have been defined to be available to his SLA Customer Group. For technical users, their permissions allow them to view all services, including those that are not yet active, but when submitting a request, they are still limited to only selecting services that are Active or In Review. Since the system filters the services a customer can view, there is no need to "build" a specialized catalog for each customer. Instead, customers in different SLA customer groups will automatically see their appropriate services based on their own linked SLAs.

Our default End User Interface demonstrates the main three methods by which customers can access services and create requests and incidents:
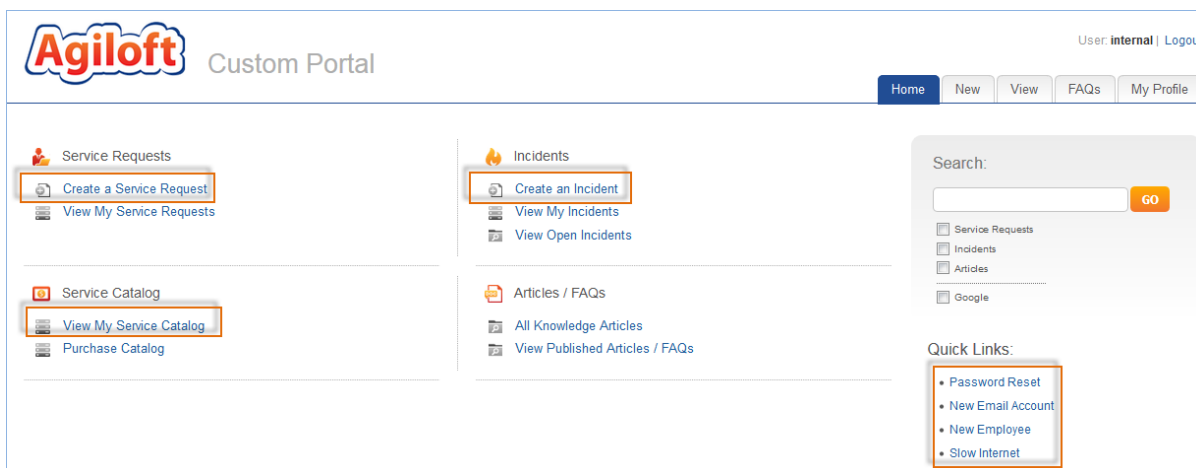

The sections below list several possible methods for presenting services.

Business Service Catalog Drill-Down Method

The first is to use the Service Catalog view of Business services, which allows the user to drill down to view the service offerings within a business service and click the "Create Request" button without knowing in advance whether the request he is creating will become an incident or a service request.   In this method, the user clicks on the "View My Service Catalog" link to see his personal service catalog. This brings up the view of the business services with a row for each business service for which they are an authorized user:
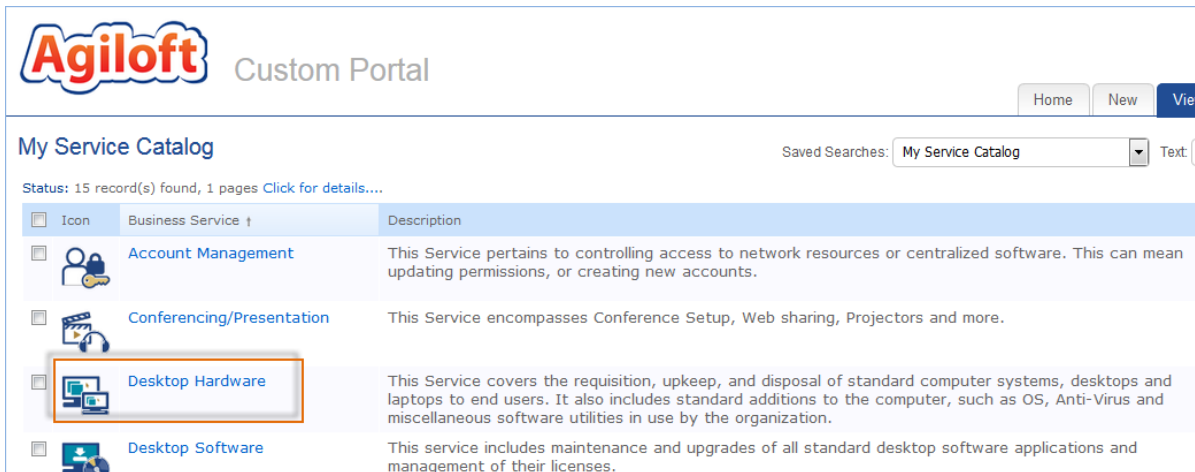
Now the user can click on a particular business service, for instance, Desktop Hardware, to view more about it and to see the individual service offerings. From the list of service offerings, the user can click on the linked service to view more information about the service offering, or they can click on the Create Request button to create a new request with the business service and service prepopulated:
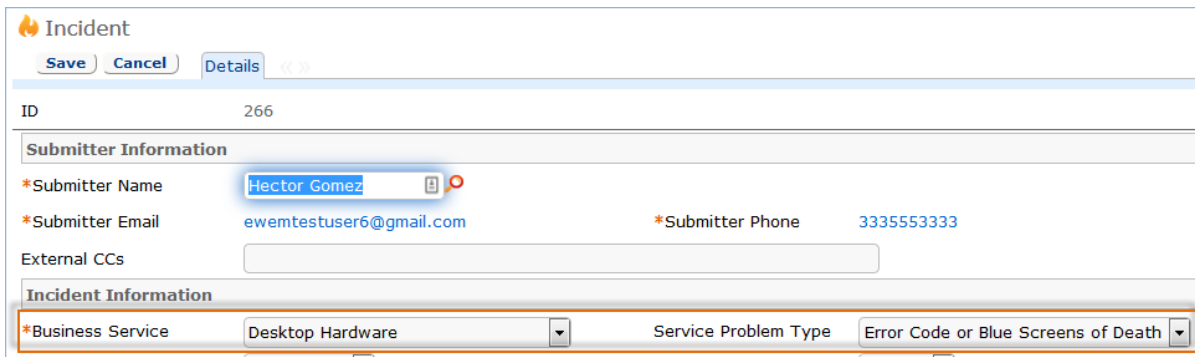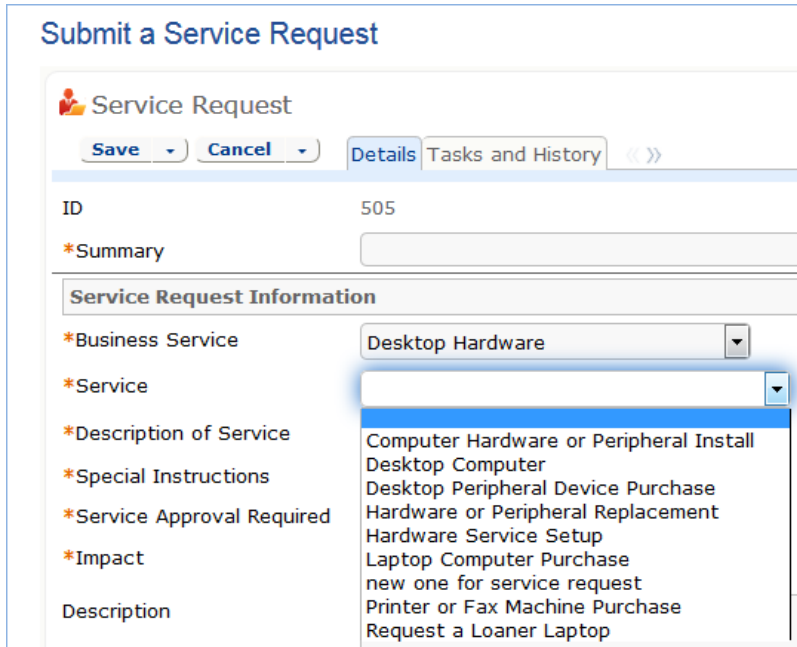

Using preconfigured hyperlinks for specific services

It is also possible to preconfigure hyperlinks that will take a user to the most popular services, or the services that are available to all users. This approach is represented by the Quick Links section on the interface,

Clicking one of these links will open up a new Service Request or Incident with the service and business service prepopulated. This is a nice method for the most common services as it only takes one click for the user to open the request. However, if your service catalog is large with many offerings, it is more cumbersome to set up and maintain.

Creating a New Incident or New Service Request Directly

Alternatively, users may create a new service request or incident directly, without having to drill down to find the services. In this case, they just open the new record and select the business service and service from the dropdowns:

This is most effective if your users have a clear understanding of which kinds of requests are handled through Service Requests and which ones are handled through Incidents. Note that if they do create a record in the incorrect table, it is easy for your staff to convert into the other.

Adding Specialized Catalogs

It is also easy to set up subsets of services based on any search criteria or fields used in the services table to make specialized catalogs available to users. In the default end user interface shown above, we have created a second service catalog called Purchase Catalog that is a subset of all services that include some kind of purchase.
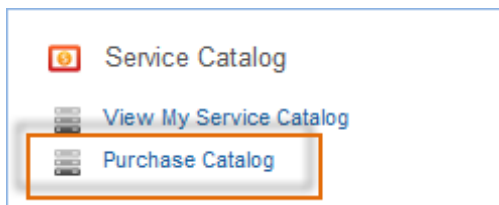
If users clicks on this item, they see those specific services and can create a request directly:

IT-Only Services

Technical users who use the staff interface of Agiloft have additional options when creating requests. They can go directly to the appropriate table in the left pane to submit a request, or they may also use the Service Manager home page and click on a link to their own service catalog. In either case, they will see additional services that were not available to customer users, such as IT Infrastructure change, based on their own SLAs and group permissions.

Showing all Services for a Customer

Within the end user records, a related table shows all the services that customer can access. In addition, reports can show which services were actually used by which customers. The report: All Services by Submitter on the Service Request table shows the services actually used by customer.

Showing all Customers for a Service

It is possible to review the customers for whom a service is available in a few different ways. Since the service is linked to one or more SLA customer groups, and users are associated with those groups, the easiest way is to drill down while in the service through the linked SLA Customer Groups. Clicking on an SLA Customer group such as the Gold / Executive group, brings up a view of the SLA Customer Group record, and from there, the Related Info tab shows all the users in that group:

It would also be possible to add a field to the service record that would show a table view of all users in those SLA customer groups directly, if needed.

Service Form Layout

We have created several sample services. Naturally, your organization will want to create its own services, delete many of our sample services, or change the associations we have made for those services. For instance, the on-boarding of new employees is currently managed through the Service Request table under the Service New Employee Setup. Some organizations may prefer to manage new employee setup through the Change Request table instead. Making this kind of change is a matter of modifying the service record and changing the field called Request Type to point to the desired table.

Since the Service Catalog is the backbone for many ITIL processes, the Services table has several special fields that are pulled into the corresponding request to enable automation, escalation, and special field visibility and dependencies based on the type of service.

Details Tab

This screen has the main details about the service. The Description of the service and the Special Instructions are shown to users when they submit a service request or incident to provide them with more information about the service.

The Attached file fields such as the Supporting Service Business Case and Service Design Package can hold supporting documents about the service, while the Service Warranty field can be used to provide further information about the expected turnaround times. The Service Criticality field represents the value of the service to the business.

Additional fields define the default team to be assigned to the request and the escalation team, as well as whether the service is billable, and if so, based on what criteria. The Auto-Assign to Responsible Team determines whether the service responsible team takes priority over the CI Responsible Team, if a CI was selected for the request, when assigning a new request. If yes, then the service team is assigned, if No, then if there is a CI and it has a responsible team defined, that team will be assigned to the request.

The Level 2 Escalation Team is automatically assigned to a request or incident when the SLA is breached, and is notified both before – at the warning interval – and after the SLA breach.

Costs for Services

Some services may be billable, and the system supports four different cost models by default, as shown in the screenshot.
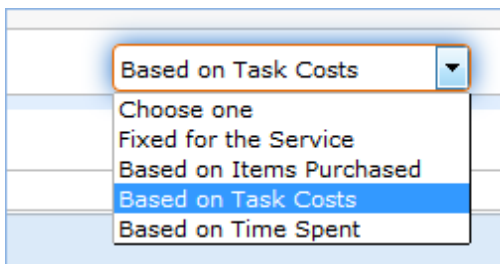
Depending on the selected cost model, the cost information is entered in the service itself, for Fixed for the service, the tasks related to the service, or it is compiled by totaling the time spent on the request or the costs for items purchased as part of the request.

When a service request or incident that is billable is completed, the Total Cost in that record is populated automatically by the system based on the entered data and the service's cost model. Reports for billable service requests and incidents can be used to export cost information to an accounting system. A Cost Center field in the request tables allows the submitter to select the appropriate cost center that should be billed for a given request.

See further information on cost handling in Service Requests Costs Handling.

SLAs / OLAs / Priorities Tab

This tab holds information about the available SLAs and OLAs for this service. The links to the service are generally made from the SLA or OLA table when those records are created and their scope defined. A service may have multiple SLAs linked to it, and the SLAs are applied to a request in priority order, described more fully in the Service Level Management section.

Service level requirements are also held in the SLA table and that is where the SLR is negotiated and defined. For a service that is being planned, the SLR may be linked to that service to provide information on the requirements that are being negotiated. Once that SLR becomes an SLA, it would typically remain linked to the service as it goes live.

Here is an example of a service for Quickbooks support with both a special SLA and an OLA:

SLAs for this Service

Active SLAs
This includes SLAs whose status is Active or In Review. SLAs are applied in descending order by SLA Type.
Status: 1 record(s) found, 1 pages. Click here to re-count records.

New | Select SLAs 🔍 Unlink   Views ▼ | 🖨 ▼

	Edit	ID	SLA Title	SLA Type ↓	Support Hours	Percentage of Breached Incidents	Percentage of Breached SRs	Number of Open Incidents	Open Incidents - Breached SLA	Number of Open Requests	Open SRs - Breached SLA
☐	📝	35	SLA for Finance team for services for Quickbooks	Service-specific	Support: 7 - 7 weekdays	0%	0%	0	0	1	0

Underpinning Agreements for this Service

Status: 1 record(s) found, 1 pages. Click here to re-count records.

New | Select OLAs 🔍 Unlink   Views ▼ | Search ▼ | 🖨 ▼

	Edit	ID ↓	Agreement Title	Agreement Type	Status	Assigned Team	Number of Tasks	Percentage of Breached Agreements
☐	📝	11	Intuit Software UC for all requests with 8-5 support	Underpinning Contract	Active	Intuit External Service Provider Team	13	7.6923%

Note that the view of the SLA and Underpinning Agreement can show the percentage of breached agreements and/or the number of open and breached requests. The next section on this tab shows the Priority group used for the service and the available priorities for that group.

Priorities Available

Priority Group: Standard Incident Priority ▼

Priority Matrix   Status: 9 record(s) found, 1 pages. Click here to re-count records.

	Edit	ID	Impact	Urgency	Priority
☐	📝	3	Company	High	1 - Critical
☐	📝	7	Company	Medium	2 - High
☐	📝	10	Company	Low	3 - Medium
☐	📝	4	Department	High	2 - High
☐	📝	8	Department	Medium	3 - Medium
☐	📝	11	Department	Low	4 - Low
☐	📝	5	Individual	High	3 - Medium
☐	📝	9	Individual	Medium	4 - Low
☐	📝	12	Individual	Low	5 - Very Low

Specialized priority groups can be created to provide a particular service with its own impact, urgency and priority values as needed. See Impact, Urgency, and Priority Management for more details on how this is done.

Approvals and Tasks Related to Services

Some services may have associated approval workflows or standard tasks or task workflows. These options are available from the additional tabs in the Service record. Approvals and tasks are available out of the box for both Service Requests and Change Requests.

Approvals for Change Requests

There are three options for how to handle approvals for a change request, shown in the Approval Generation Method drop-down below: using a predefined approval workflow, manually created ad hoc approvals, or both. For **a Predefined Workflow Only, the user will select from existing approval workflows related to change requests. If none yet exist, they can use the button to create a new approval workflow on the fly.**

If the user selects **Manually Created in Request Only then** no further information is needed in the service record, and they will be able to add the approvals manually within the individual change request.

Approval View in the Change Request

The selections made in the Service will result in different options appearing within the Change Request.

In a change request based on a predefined workflow only, the user will see the workflow listed and a button to generate the approval records. If the user has permission to edit the Approval Workflow Title field, they would also be able to change to a different workflow as needed. If both predefined and manual approval creation are permitted, the user will see an additional checkbox, Create Approvals, that can be expanded to add one or more ad-hoc approvals. If only the manual approval creation is defined, then the user will just see the option to Create Approvals and will add any approvals needed.

 Note that if the Approval Required field in the service has a Yes value, at least one approval will need to be created and approved to move the change request forward to completion.

Tasks for Change Requests

The Tasks tab of the service allows the selection of a method for task generation. The available methods depend on the request type and they are different for service requests and change requests. Change Requests have two options for the task generation method: Predefined Task Workflow and CR Single Task from Template.

Change requests are typically associated with one or more configuration items. Task handling for change requests is therefore designed to create a task for each configuration item that has been associated with the request. If the user chooses 100 configuration items for a change request and generates tasks, whichever tasks are defined for the change request are created 100 times, one for each configuration item. The tasks are generated in this way by using a task template and pushing its details down through the configuration item records, so this necessitates restricting some of the other task generation options.

Predefined Task Workflow - allows the selection and/or creation of a Task Workflow with predefined task templates arranged in a sequential or parallel order. See more details about configuring task workflows in the Task Workflows section.

Note that if your workflow has 3 tasks, they will each be generated for each configuration item.

The **CR Single Task from Template o**ption is more common for change requests involving numerous configuration items. Here you define a single task template to be used with the change request.

Note that a single task may have several predefined steps, presented as a checklist, to assist technicians in performing the task. See the sections on Task Templates and Task Steps below for more details on how to set this up. Below is an example of a task template that has been defined with specific steps:

Task Steps

Use Task Checklist   ◉ Yes ○ No

Create New Checklist
Item?   ☐.

Status: 5 record(s) found, 1 pages. Click here to re-count

Mass Edit | Save Changes   Cancel Changes

☐ Edit	Step Number ✏ ↑	Step Name ✏
☐ 🖉	1	Perform Backup
☐ 🖉	2	Stop all Services
☐ 🖉	3	Install Microsoft Patches
☐ 🖉	4	Reboot Server
☐ 🖉	5	Check Services are running

Here is how the setup above will appear in the task that is assigned and updated by the user doing the work:

Configuration Item Details

Configuration Item	server2 🔍		CI Operational Status Working

Working Notes

Task Steps	☐ Perform Backup
	☐ Stop all Services
	☐ Install Microsoft Patches
	☐ Reboot Server
	☐ Check Services are running
Working Notes	

Using task steps within a single task makes good sense for a change request. While setting up the service, if the task template you want to use does not yet exist, the Create and Apply New Task Template button can be used to bring up the task template screen to define a new template, and once saved, that template will be selected for the service you launched it from:

Default Task Template

Create and Apply New Task Template

In the actual change request, it is possible when using this task method to select more than this one task template to generate. Once having generated the default task, the user may select from other predefined tasks and generate them as well.

Tasks for Service Requests

For service requests, there are some different task generation options. There are three possible methods:

- **Predefined Task Workflow** – this is the same as the usage in change requests, except that the tasks are not tied to CI's and are just created once per request. With this choice in a service request's service, another field - Enable Ad Hoc Tasks – appears, and if it has a Yes value, then in addition to the predefined tasks, the user may create additional tasks for a particular request.
- **User Selected Tasks** –allows the service manager to create a set of possible tasks that will be displayed in a service request as checkbox items, so the user may select which tasks are relevant for this particular service request and generate just those tasks:

The Create Task Template button is used to create the potential tasks as templates. Ad hoc tasks may also be enabled for this type by selecting Yes for the Enable Ad Hoc Tasks? field shown above.

The result of the service shown above in a given service request:

The user checks the boxes for the tasks he wants, and clicks the Generate Task(s) button to generate the individual tasks. These tasks, since they are all optional, are defined as parallel tasks without specific prerequisites.

User Generated Ad Hoc Tasks – this selection allows the user to create manual tasks in the service request without reference to any preexisting templates. In this case, the user will see a button in the service request to create a new task. Ad hoc tasks can be made sequential during creation by selecting one or more prerequisite tasks as the individual tasks are created.

Performance Tab

The performance tab on the layout is used to hold information about the number of open incidents or service requests for a service and the percentage of those that have breached their SLA.

Depending the request type, it shows different fields: a related table of open incidents, or a related table of open service requests, and the relevant statistical fields:

These values can be shown in reports on the service table as well as on the Incident or Service Request tables.

Service Provisioning Tab

This is where the workflow for on-boarding a new service is done. There are fields to track communications, updates, knowledge and other materials that may be relevant, and automation to make it easy to email people responsible with information or questions about the service.

New Service Onboarding and Management

In addition to the notes and links to files and knowledge, the Status field is used to manage the onboarding process and to define the subsets of the Service Portfolio. It has default values of:
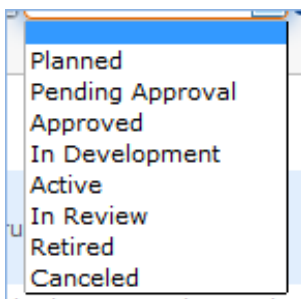
- All services that are in a status of Planned, Pending Approval, Approved, or In Development are shown in the **Service Pipeline** list and report.

- Those with a Status of Active or In Review are shown in the **Service Catalog** list.
- Those that have been set to Retired are shown in the **Retired** list and report.

> On the date of the **Next Review Date** the Status is automatically changed to In Review and the Service owners are notified by email. This is to cause every service to be re-evaluated on a regular basis.
>
> - During the review, if the service owners decide the service is still valid as it is, they can set the Status back to Active.
> - If they decide to retire the service, they can set its Status to Retired.
> - If they decide to work on a new revision of the service, they can clone it using the "Clone" button and make revisions and changes in the new record while keeping the old version's status of In Review. Once the new version is ready to activate, they can set the Status of the preceding version to Retired.
>
> Note: While the Status is In Review, the service is still available to users.

Once the Service Catalog is initially set up by creating the appropriate services in the Service table, new service items will typically be created by staff members only when a new service is planned or initiated. Initial permissions are set so only users in the admin, Service Manager, and Change Manager groups can create, modify, and delete services.

Ownership

Services are owned by the person who creates the service.

Workflow

The following states and state transitions are permitted in this table:
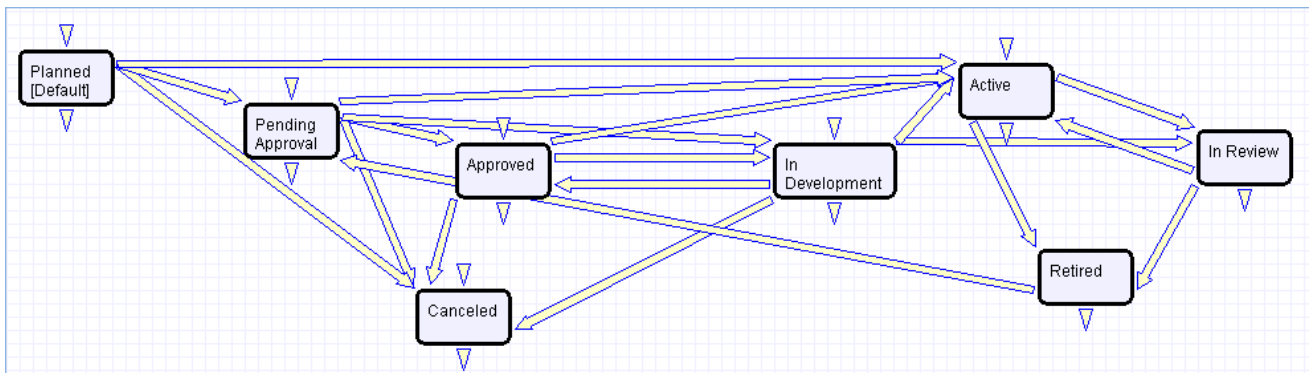
Saved Searches

Some of the default saved searches provided are detailed below:

Saved Search Name	Search Description
Service Pipeline	Status is Planned, Pending Approval, Approved, or In Development
Service Catalog	Status is Active or In Review
Retired Services	Status is Retired
Planned Services	Status is Planned

Reports

Report Name	Description
Service Catalog	This report shows a chart and list of all active and in review services, the services that are available to customers for requests:
Service Pipeline	This report shows all services in the pipeline.
Retired Services Report	This report shows all services whose status is Retired.

Service Level Management

This section describes the tables that control the functioning of service level management in Agiloft. There is additional discussion about how SLM is applied throughout the request tables, particular in the sections on Service Requests and Incidents.

The main tables related to Service Level Management are:

Table name	Description
SLAs	This table holds a record for each negotiated Service Level Agreement between the IT department and its customers. SLAs are made available to one or more SLA Customer Groups. Each user is assigned to an SLA Customer Group and that defines the set of services available to that user. For more information, see Service Level Agreements.
SLA Targets	This table holds a target record for each combination of SLA, Priority, and Request type that defines the expected Response Time, Warning Time, and Resolution time. Targets are applied to specific service requests or incidents based on the SLA and the priority of the request or incident, and then are used to manage escalation and notifications. Each SLA has multiple targets.
Underpinning Agreements	This table holds a record for each Operational Level Agreement or Underpinning Contract between the IT Department and individual internal teams or departments and/or external service providers. These records detail the scope of the service to be provided, the team responsible for providing the service, and any specific task templates to which the OLA or UC will apply. It will also have multiple OLA Targets, for every combination of priority available for the tasks to which the Underpinning Agreement will be applied. For more information, see Underpinning Agreements.
OLA Targets	This table holds a target record for each possible priority for the tasks that will be assigned to an UA. The OLA Target will be applied when tasks are created and assigned to set the due date, warning date, and response date for the task, so the performance can be measured against that target.
SLA Customer Groups	This table holds a record for each set of users who will have specific SLAs negotiated on their behalf. All customer groups are linked to any SLAs whose Type is Corporate, while Service-Specific or Customer-Specific SLAS may be created and assigned to just one or more SLA Customer Groups.
Surveys	Surveys are sent out with every service request and incident with a very simple questionnaire and the information is gathered to assist in measuring service quality.
Service Improvement and Quality Plans	This table holds records for service improvement plans and service quality plans. These records can be linked to the CSI register and to customer surveys for further support. For more information, see Service Plans and CSI Register.
CSI Register	The CSI register acts as a log for gathering service improvement information and opportunities. CSI records can be linked into the Service Improvement and Quality Plans. For more information, see Service Plans and CSI Register .

Service Level Agreements

The SLAs table manages the service level agreements between the IT department and its customers. SLAs define specific SLA targets for response and resolution times for incidents and service requests. These SLA targets also define diagnosis and resolution times for problems. SLA targets are based on the priority of the request record and are applied or updated whenever the priority is set or changed.

Each SLA has its own SLA targets.

The SLA structure is designed to be extremely flexible and to handle situations in which each department has completely different response times from other departments, multiple sets of services have different response times, or a combination of the two.

When applying an SLA to a request or incident, there are two factors that determine which SLA is applied: the SLA types and the SLA Customer Group of the submitter.

SLA Types

There are four different types of SLAs and they are applied to new service requests or incidents in priority order as follows:

Configuration Item - if a configuration item has its own SLA, this takes priority over other SLAs, for example an SLA for a critical configuration item or class of items

Service-Specific - This is an SLA for one or more specific services with typically reduced time frames for its SLA targets that may be available to only some customer groups, for example an SLA for a high priority service that should have better response times than less business critical services, or for single services that should only be available to a specific groups of users

Customer Group - This is an SLA that typically covers multiple services but that applies to just one specific Customer Group, for instance an SLA for the finance manager group that gives them better resolution times for mission critical financial-related services

Corporate - This is an SLA is for multiple services that are made be available to all users in all customer groups, typically at a base level of service, for services that may not be included in the other SLA types

SLA Customer Groups

SLAs are defined to be available to one or more specific SLA Customer Groups. Corporate SLAs are applied to all groups automatically. Both Service-Specific and Customer Group SLAS can be applied to one or more customer groups. So a customer group might have a combination of all three types of SLAs available to it, and all three might even be linked to some overlapping services.

Users of the system each belong to one SLA Customer Group, and that group defines the list of services that user can request, based on the SLAs tied to his group and the services tied to these SLAs.

> **Example**
> A user could have one corporate SLA available to him, one service-specific SLA available to several groups including the sales group, and one Customer Group SLA negotiated just with the sales group that covers some specific services. They will be able to submit requests and incidents for any of the

services linked to any of these three SLAs.

If they submit a request for the one service that is covered by all three SLAs, then the service-specific SLA will be applied based on the priority order described above. If they submit a request that is covered by the corporate and the customer group SLA, then the Sales Customer Group SLA will be applied. If they submit a request that is only covered by the corporate SLA, then that SLA will be applied.

However, in all three cases above, if the request was for a configuration item that had its own SLA, that SLA would be applied instead. If there are services outside those three SLAs, he will not see them or be able to submit requests for them.

SLA Layout

It is important to set up the SLAs carefully to ensure the correct coverage for all services - typically in a corporate SLA - with the special coverage for certain services or certain groups within the organization. SLAs may be opened by clicking on the Edit icon in the left pane. They may also be opened from within a service record on the SLAs tab:
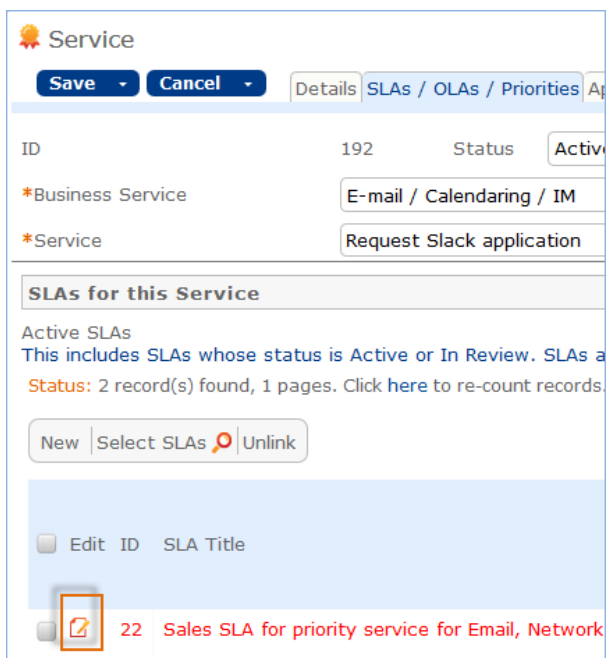
The Main Details tab of the SLA contains the primary fields. SLAs may be applied to both incidents and service requests or only to one request type. The multi-value Applies to SLA Customer Groups field is used to define the groups that can use the services in this SLA.

Fields for Scope and attached files gather the negotiated data and requirements for the SLA. The SLA Owner Team defines the internal IT team that is responsible for managing the SLA, while the availability and support hours define the hours that will be used for SLA target measurement and reporting.

All SLA targets are measured against the Support Hours for the SLA, so it is critical to set these up correctly.

ID	35	Status	Active ▼
*SLA Title	SLA for Finance team for services for Quickbooks		

SLA Details

*SLA Type	Service-specific ▼	Available for Request Types	Incident, Service Request 🗒
*Applies to SLA Customer Groups	Finance Managers 🗒		
Description	This offers services related to Quickbooks only. It is available only to the Finance team.		
Attached Requirement Documents	Attach/Manage Drag&Drop files	Attached Service Level Agreement	Attach/Manage Drag&Drop files
Scope	It provides access to special services and incident types.		
IT Only Notes			
*SLA Owner Team	Service Management Team ▼		
Service Availability Schedule	Available: 24 x 7 except Sat. 8pm to midnight ▼		
*Support Hours	Support: 7 - 7 weekdays ▼		

Support Hours

The support hours for SLAs, as well as the availability and maintenance schedules for requests and configuration items, are defined in Agiloft as special teams. Teams can have specific working hours and the system is able to include or exclude team working hours in any kind of time measurement, for elapsed time fields or reports, so this gives the ability to set targets based on such team hours and then measure them.

For example, here are the working hours defined for the Support: 7 - 7 weekdays team:

```
⦿ As defined below
Sun  ---------  ▼  -  ---------  ▼
Mon  07:00      ▼  -  19:00      ▼
Tue  07:00      ▼  -  19:00      ▼
Wed  07:00      ▼  -  19:00      ▼
Thu  07:00      ▼  -  19:00      ▼
Fri  07:00      ▼  -  19:00      ▼
Sat  ---------  ▼  -  ---------  ▼
```

Several default teams have been set up with and named for specific working hours, but if you need to manage other kinds of schedules, you can modify these teams or create new ones. The Teams table has a special custom field to filter dropdowns for support hours, maintenance hours, and availability hours, as well as for the drop-downs for assigned teams within the request tables. This field is called Team is Responsible For.

When setting up new teams for this purpose, it is important to check the appropriate box so the team will appear in the various dropdowns where it belongs.

Services for an SLA

The Services tab allows the user setting up the SLA to select the business services, and then the related service offerings to which the SLA will apply. The user can use the Add All Services button to add all services underneath the selected business services above, or may use the lookup to select individual services filtered to those for the selected business services.

Configuration Item SLAs

If the SLA type is configuration item, instead of setting up specific services, the user will link the SLA to one or more existing configuration items:

An SLA can also be selected when creating or editing a Configuration Item record. Here is the corresponding configuration item record:

Below the services section is the embedded table of the SLA targets that will be used for this SLA. When setting up a new SLA, click the New button above this table to create new SLA target records.

SLA Targets

The SLA targets table is used to define the turnaround time for a service request, incident, or problem record. Each target is linked to one SLA and is shown within that SLA. Since an SLA may cover more than one request type, the targets can be distinguished by the request type as well as the SLA. The SLA target form has the following fields:

The key elements are the Priority value, the Request Type, and the three time fields that set the response, warning, and resolution times. Each of these times is measured only during the defined support hours for the SLA. So if the SLA has hours from 7 AM to 7 PM and a service request with the above target is submitted at 6 PM on Monday, the response time will be due at 7 PM, the warning time will occur at 2 PM Tuesday (with 1 hour remaining Monday and 7 hours starting from 7 AM Tuesday), and the resolution time will be set to 3 PM Tuesday.

Applying SLA Targets in Service Requests and Incidents

When a service request or incident is created, the appropriate SLA is applied based on the criteria described above. Then based on the Priority of the record, the appropriate SLA target is applied and linked into the request or incident.

When the SLA target is linked, if no approvals are needed first, a rule sets three date fields in the request: the SLA Due Date, SLA Response Due Date, and SLA Warning Date by adding the values set in the target for each of these times to the current date/time plus that value according to the working support hours for the SLA.

In a case where there are approvals before work can begin for a service request, the SLA Response due date is set immediately, and the SLA Warning Date and SLA Due date are set when all approvals are completed and the status is set to Approved. In that case, the time is measured from the Date Approved rather than the date the record was created.

Available Priorities

In order to assist in setting up SLA targets, the available priorities for the various request types included in the SLA are shown in the Available Priorities table view. If there are only 3 priorities for service request, but 5 priorities for incidents, that is the default setup, then if an SLA covers both request types, it should have 8 SLA Targets, one for each possible priority - as shown above.

Related Information Tab - Measuring Performance

The Related Information tab holds information about all open service requests and open incidents that are based on this SLA. It can be used in reporting to show the current situation by SLA.

Emails and History Tab

The Emails and History tab hold the usual communications table showing all emails sent about the SLA and the standard History fields showing the Date Created, Date Updated, and all other changes made to the SLA record.

Onboarding an SLA - Status Workflow

The SLA record has several workflow statuses to manage the on-boarding of new SLAs. The SLA on-boarding process may well begin as a Service Level Requirement - the first Status allows the SLA record to be created in a status of Planned - Service Level Requirement, so that the necessary requirement information can be collected and gathered in this record during negotiations. From there it may move to In Negotiation, and when ready to use, Active.

When SLAs are applied to service requests and incidents, only those in a Status of Active or In Review may be applied. If an SLA is replaced by a new version, the old one can be set to a Retired status, and a planned SLA may be canceled if it does not meet the needs of the organization.

Date Fields for Tracking Progress

When an SLA changes to Active, the Date Activated field is automatically populated. In addition, the Next Review Date can be set to trigger a reminder to review the SLA and see if it should be adjusted, at whatever interval is ideal. SLAs are automatically set to a Status of In Review when the Next Review Date arrives and an email is sent to the SLA Owner Team notifying them that the SLA should be reviewed. This email is sent by a time based rule that checks each day to see if the current date matches the next review date.

The Ongoing Notes field is an append only field that will date/time/user stamp all updated comments.

Reporting

Reporting for SLA performance is done from a variety of places.

Excel dashboards have been created for the Service Request and Incident tables that allow the data to be sliced by any factor, from the individual SLA, to the individual business service or services, by assigned team and so on.

SLA Dashboard for Incidents

The report on the Incident table is shown here:

☐	✎	🔍	SLA Dashboard		requested and still pending,...	No	Jul 05 2016 20:03	Excel	Standard

When clicked, it results in an Excel file that can be manipulated to show any statistics related to SLA metrics:
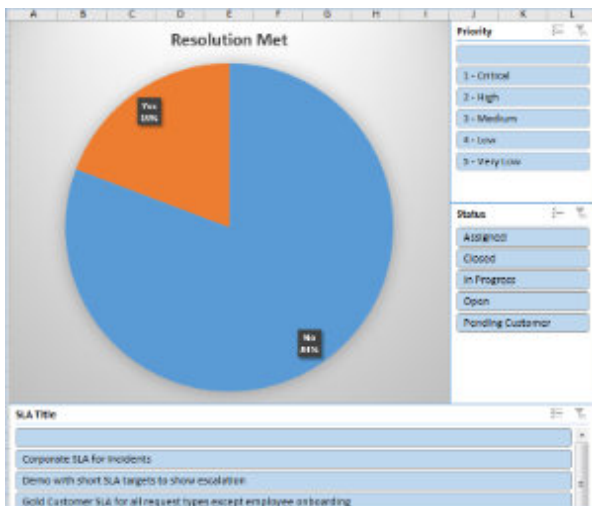
Service Request SLA Dashboard

The same Excel-based report runs on the Service Request table.

✎	🔍	Service Request SLA Dashboard			Yes	Jul 05 2016 20:07	Excel	Standard

It shows a similar overview with slicers for each factor:
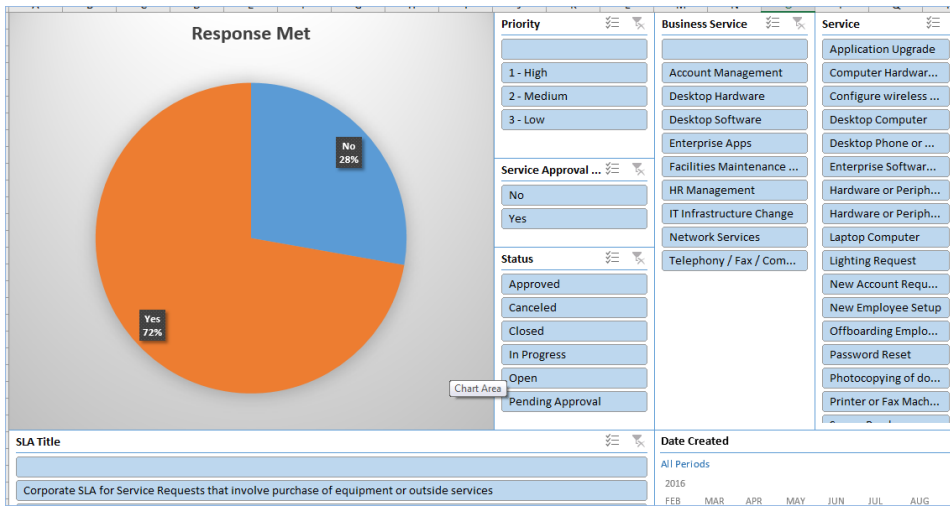
Open Service Requests that have breached their SLA Resolution Time

This report runs on the service request table and also appears in the Service Management Dashboard:
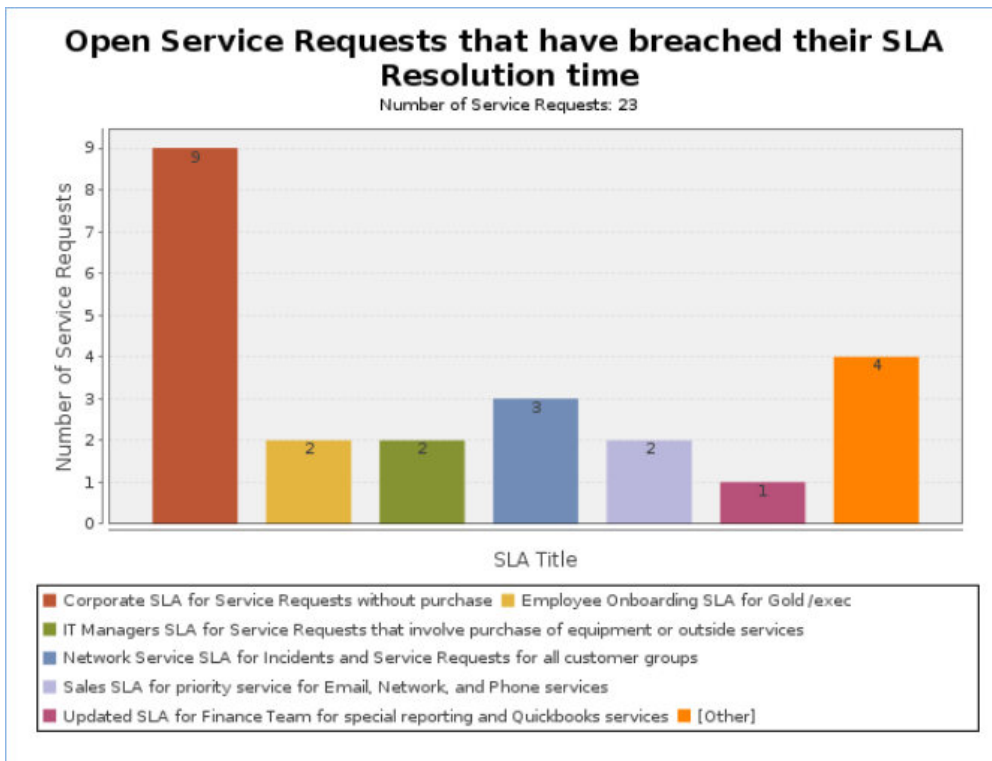
Active Incidents that have breached SLA by Priority

This report on the incident table shows the open incidents that have already breached their resolution time. It also appears in the service management dashboard.

Every chart has the drilldown capability through an HTML tab to go directly to the underlying incidents.
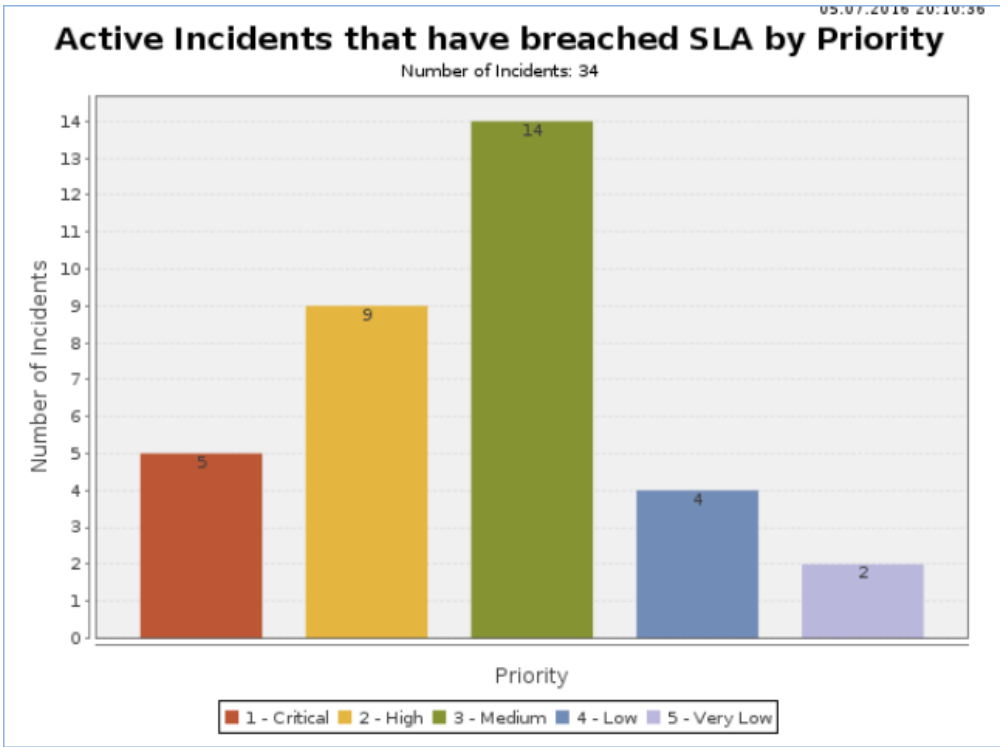
Closed Service Requests: Percentage meeting SLA in past 12 months by assigned team

This chart and report shows by team the percentages of requests that met the SLA.

Service Manager Dashboard

The dashboard is accessible to service people to quickly find incidents and service requests that have breached their SLAs and to drill down to work on them:

Automatic Performance Monitoring

The Services table has rules monitoring it to keep service records updated with the current number of open requests, and to set the color of the service to red when some threshold is surpassed. A time based rule tracks the performance of all services with regard to current status of breached incidents and sets the row coloring for service based on the result:

- If the percentage of breached incidents or service requests is greater than 30% the row color is set to red.
- If the percentage is between 15-29.99% it is set to orange
- If the percentage is lower than 15 it is not colored.

	Edit	ID ↓	Service	Number of Open Incidents	Open Incidents - Breached SLA	Percentage of Breached Incidents	Number of Open Service Requests	Open SRs - Breached SLA	Percentage of Breached SRs	Request Type	Responsible Team
☐	✎	203	Demo: building is on fire!!	5	5	100.00%	0	0		Incident	Risk Team
☐	✎	192	Request Slack application	0	0		2	0	0.00%	Service Request	Desktop Applications Team
☐	✎	191	Quickbooks Problem	8	7	87.50%	0	0		Incident	Intuit External Service Provider Team

Event Creation for SLAs with Performance Problems

SLA performance issues can result in the generation of an Event in the Event Management module. There is a weekly rule that searches for all SLAs whose Percentage of breached incidents is greater than 75% and generates an event for each one. These events will notify the service managers involved so they can analyze the reasons for the breach and take appropriate action.

Rule: TB Weekly: Create Event if percentage of breached incidents is higher than 75%

Converts to an event with the appropriate information filled out.

Underpinning Agreements

The Underpinning Agreements table holds records for all Operational Level Agreements and Underpinning Contracts negotiated between the IT department and specific internal teams or outside service providers.

These agreements define the main contact information for the other party, the support hours and services and tasks that are covered by the agreements.

Agreement Layout

The tabs and fields of the Underpinning Agreements are described below:

Details Tab

The Details tab holds most of the fields for the Underpinning Agreements. It is very similar to the fields for SLAs. It defines the request types that will be covered and the type of Agreement (OLA or Underpinning Contract). It includes description and scope fields to define what the agreement will cover. It defines the team that owns the agreement, as well as the team that will be assigned to tasks related to the agreement. It also defines the support hours that will govern time measurement for tasks related to the agreement.

Just as for SLAs, the Date fields are used to indicate when the agreement was made active - set automatically to the date the Agreement status changed to Active - and to trigger the Next Review Date.

Underpinning Contracts

If the Agreement Type is Underpinning Contract, there are some additional fields to identify the Vendor and contact information, as well as to have a link to the Contracts table, which may hold the details of the contract with the vendor.

Agiloft's Company table is used to hold all service provider and other companies, while the External User table holds contacts outside your company. See Companies Table and People Table, Employees and External Users Sub-Tables for more details on these background tables. The contracts table is a fully functioning contract management system and can be used to manage and store any service provider contracts. See Contracts Table for details on contract management out of the box.

Services Tab

The Services tab is used to link the agreement to specific Business Services, and from there to specific service offerings linked to those business services. It is also where the OLA targets are defined.

OLA Targets

The OLA Targets table is used to define the resolution target times for tasks related to an OLA. The difference between OLA Targets and SLA Targets is that they do not apply to a whole service request or incident, but to specific tasks within that request that are predefined to be covered by an OLA or UC. Therefore there is an extra step to set up the targets, which involves linking them to specific task templates to which they will be applied.

Just as for SLA Targets, the OLA Target is also tied to, and applied based on, the priority of the request or incident. Here is an example of an OLA Target for the Purchasing Team task for employee on-boarding requests:

Tasks Tab

The Tasks tab displays the task templates to which the OLA will apply. Task templates are created as part of a task workflow once the OLA has been created, and they are linked to the relevant OLA, which causes them to appear here. Also on this tab are all actual Tasks done under this Agreement, with some statistic fields measuring the number of breached agreements and percentage of breached agreements. This allows the effectiveness of all OLAs to be reported on from the Underpinning Agreements table as well as from the Tasks table.

History Tab

The History tab shows dates fields and the history of all changes to the record.

Reports

The default table view provides a quick summary of the effectiveness of all Underpinning Agreements:

Underpinning Agreements

Status: 6 record(s) found, 1 pages. Click here to re-count records.

New | Mass Edit | Delete | Actions ▾   Views ▾ | Search ▾ | 🐱 ▾ | 🔍 ▾ | 🖨 ▾

Edit	ID ↓	Agreement Title	Agreement Type	Status	Assigned Team	Number of Tasks	Percentage of Breached Agreements
☐ ✎	14	Purchasing Team OLA for New Hire Requests	OLA	Active	Purchasing Team	32	3.125%
☐ ✎	13	Facilities Team OLA for New Hire Requests	OLA	Active	Facilities Team	33	6.0606%
☐ ✎	12	System Administration Team OLA for New Hire Requests	OLA	Active	System Administration Team	66	4.5455%
☐ ✎	11	Intuit Software UC for all requests with 8-5 support	Underpinning Contract	Active	Intuit External Service Provider Team	13	23.0769%
☐ ✎	8	Security Team OLA for Service Requests with 24x7 Support	OLA	Active	Security Team	2	50%
☐ ✎	7	Security Team OLA for Service Requests with 8-5 support	OLA	Active	Security Team	2	50%

Agreements by Team Report

This report shows the average percentage of breached agreements by the assigned team:
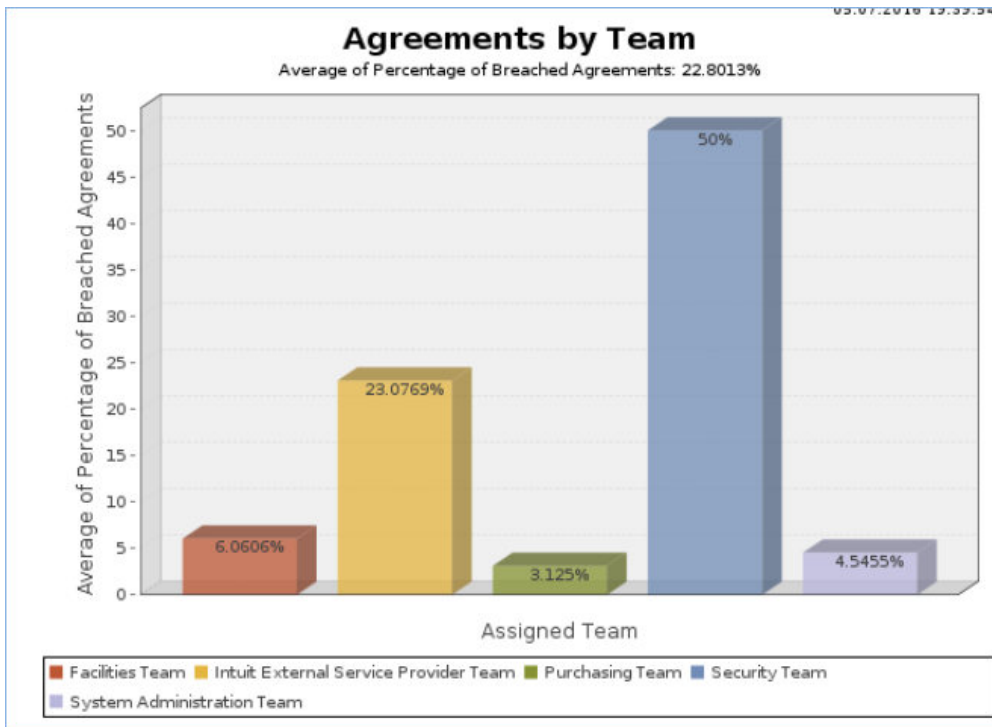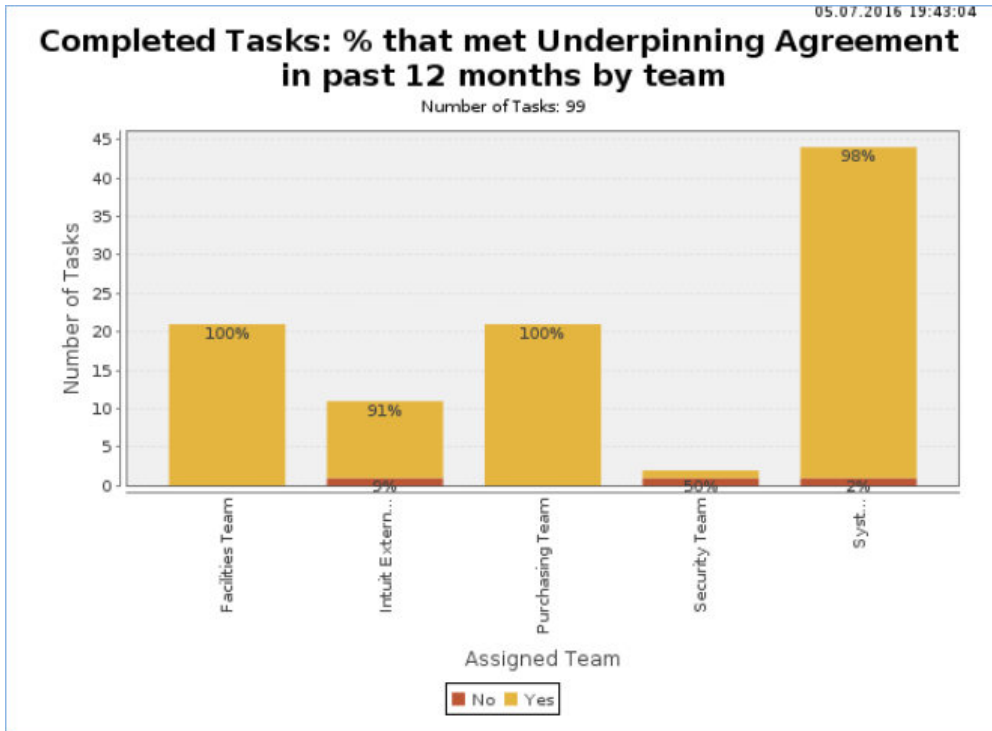
Service Plans and CSI Register

The Service Plans table is used to gather information for Service Improvement Plans and Service Quality Plans and to manage implementation of service improvements. The layout is similar for both types of plan:

The plan can be linked to one or more CSI Register records and has a status field for tracking progress. In addition, customer surveys can be linked into the plan to provide customer feedback and take that into account.

CSI Register

The CSI Register is a simple table for logging Continual Service Improvement suggestions and opportunities. These can then be linked into a Service plan as needed. The layout includes the following fields:

ID	3		Status	Submitted ▾		
*Summary	Improve callback response time for urgent issues					
Service Owner	Chris Support🔍					
*Submitted By	Admin CP🔍		*Submitter Email	chris.pankaew@agiloft.com		
Description	We have received various complaints about the callback response time not being fast enough for urgent issues (i.e. Priority 1). We need to implement measures that will help us funnel and address these types of issues more quickly, and to treat them more differently than other more normal issues in the pipeline.					
Attached Files	Attach/Manage					
	Drag&Drop files					
Working Notes						

Linked Services

Status: 3 record(s) found, 1 pages. Click here to re-count records.

Select Services 🔍 Unlink   Views ▾ 🖨 ▾

	Edit	ID ↓	Service	Business Service	Request Type	Status	Approval Required?	Are there tasks for this Service?	Billable
☐	📝	169	404 or Not Found Error to Company Site	Web Services	Incident	Active	No	No	No
☐	📝	168	Company Website Problem	Web Services	Incident	Active	No	No	No
☐	📝	135	Account Locked	Desktop Software	Incident	Active	No	No	No

Specific services can be linked to the CIS record and details can be added in the Working Notes append only field. Any Service improvement plans can also be linked in or the link created from the Service plan displayed here:

Service Improvement Plans

Status: 1 record(s) found, 1 pages. Click here to re-count records.

New | Mass Edit | Delete | 🔍 Unlink | Actions ▾   Views ▾ | Search ▾ | 🔔 ▾ | 🔍 ▾ | 🖨 ▾

	Edit	ID ↓	Status	Plan Type	Priority	Plan Name	Business Service	Submitted By	Attached Files
☐	📝	2	Submitted	Service Improvement Plan	Medium	Example Plan to improve Email service	E-mail / Calendaring / IM	Admin JMR	

Configuration Item Management

This section contains information about the setup of Configuration Items and all of their relationships and related tables. The primary tables for configuration management are:

Table name	Description
Configuration Items	This is the main table holding a record for each trackable configuration item. It has relationships to the request tables, the Knowledge Management tables and to tables that provide the CI Subtypes, Catalog items, CI Relationships, and so on. See Configuration Items Table for more information.
Catalog Items	This table holds records for standard models or services that may be used in multiple individual configuration items. It is used to provide consistent descriptions and model information content within a configuration item. See Catalog Items Table for more information.
CI Subtypes	This table holds a record for each configuration item subtype. Each record is linked to a particular CI Type. The subtypes are pulled into the other tables, such as Catalog Items and Configuration Items, for consistency. Subtypes may or may not be used in Catalog Items.
Items Requested	This table holds a record for each item that is requested by a customer or staff members to be purchased or pulled from in stock items. These records are created as part of creating a service request that involves equipment or service purchase. The Items Requested are shown within such Service Requests and the records are processed by purchasing or the configuration management team. See Items Requested Table for more information.
CI Relationships	This table defines the relationships between CI's, so that they may be displayed in diagrams to make the dependencies of components clear. Any CI may be related to one or more other CI's in a specified kind of relationship, and all such relationships can be displayed in a diagram format. See CI Relationships Table for more information.
Relationship Types	This table holds pairs of records for each relationship type that may be used to characterize the relationship between configuration items, such as Impacts/is Impacted by or Uses/Is used by. It is used within the CI Relationships table to create mirrored pairs of relationship records when a single record is created. See Relationship Types Table for more information.
Baselines	The Baselines table allows a configuration baseline to be created for a CI and stored for future reference or to be used to roll back unsuccessful changes. Baselines are related to individual configuration items. See Baselines Table for more information.
Definitive Media Library	The records in this table are used to store information about the physical or electronic location of copies of the code for software applications.

Configuration Items Table

The Configuration Items table holds records containing information about your company's Configuration Items (assets). It may include CI's that are in inventory and not in service, as well as all assets that are in service. The Configuration Item fields are described in the Configuration Item Fields Table in the Appendix.

Configuration Item Creation

Staff may add a Configuration Item manually to the system by creating a new record, or integrated Discovery systems may create CI records automatically with specified attributes. Configuration Items have type-specific attribute fields (Manufacturer, Model, Serial Number, etc.) and relationships to other CIs, Service Requests, Problems, and Change Requests.

Related Service Requests, Problems, and Change Requests are displayed as lists of records within a CI, so the history of a given CI is readily available. Configuration Items may also be related to other CI's in an upstream and downstream embedded table.

Record Creation

Configuration Items may be created independently or when selecting Configuration Items in Change Request and Contract records. Configuration Item Name and Configuration Item Type are required fields.

Configuration Item Tabs and Fields

The tabs and fields in the Configuration Items table are described below:

Details Tab

The Details tab contains general information about the CI's status, importance, SLA, responsible team, definitive media library, knowledge articles, and other details. The icon next to the Relationship Diagram field can be clicked to show a graphical display of the relationships between this CI and others. For instance, the following CI for Email/Calendaring/Slack has the following relationships as shown in the diagram:
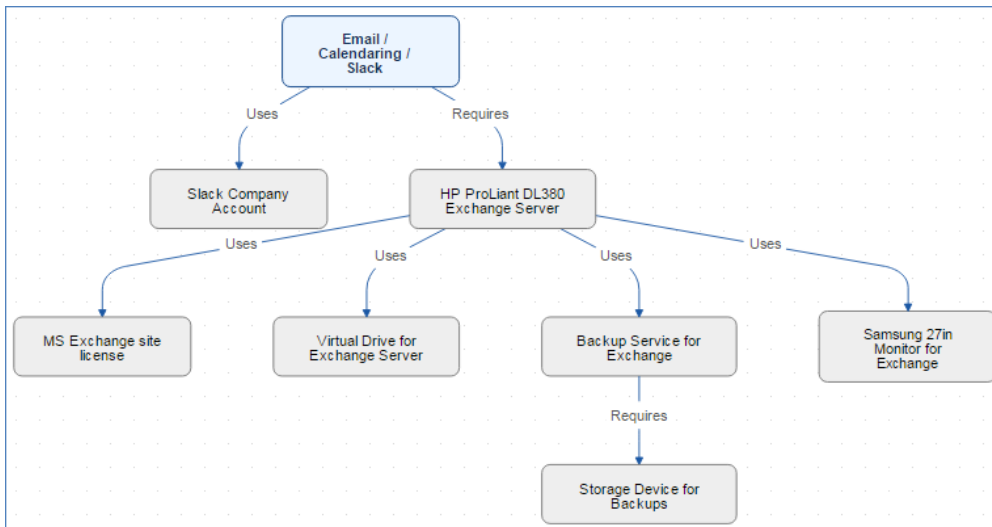
The options for Status are:
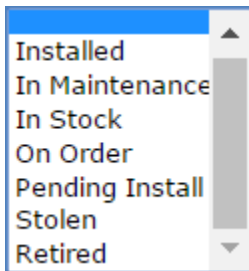
CI Type can be set to any of the following options:
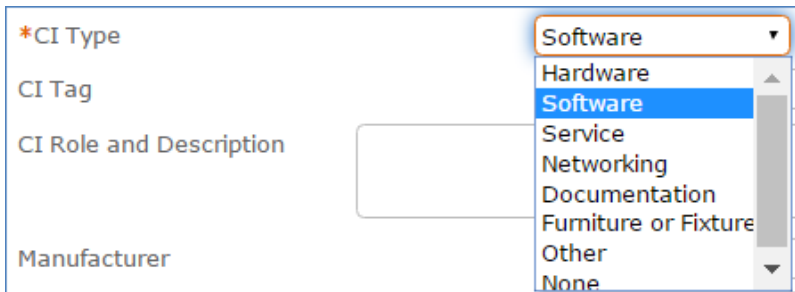
For each CI Type that is selected with the exception of None, there is a list of relevant CI Subtypes to choose. For example, the CI Subtypes for the CI Type of Software are:
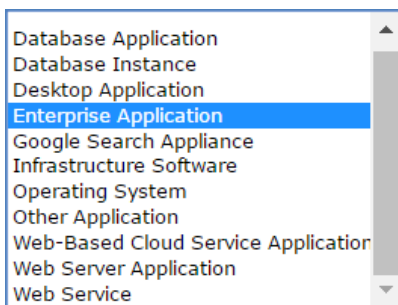
Additional information that can be entered for a CI includes the CI Tag, Serial Number, CI Role and Description, Manufacturer, Vendor, Item Name/Number, and Item Description. Data sheets or files containing prices can be attached to the Data Sheets and Pricing Info field, and any other files can be stored in the Attached Files field.

Requests / Work Done Tab

This tab contains information about related incidents, service requests, problem requests, change requests, releases, and tasks done.

Usage / Relationships Tab

The set of fields in this tab can be used to establish relationships between multiple CIs, and to specify the location of the CI.

In this example, the firewall server has three types of relationships with other CIs: one in which this firewall server impacts another internet service CI, one to indicate that this firewall server uses a router, and one showing that this firewall server is contained within a network data center.

Changes / Events Tab

This tab contains information about any change freezes, the CI change log, baselines, and related events. All events for that configuration item are shown. The Change Freeze Information section can be used to indicate that changes should not be made to this CI within a certain period of time. User permissions to edit these records can then be filtered to only allow editing if the CI is not frozen

Agiloft can be integrated with an asset polling system, and any changes made through this integration are captured automatically with change logs. The information tracked includes the date/time the changes were made and the field values before and after the change. This information is stored in a change log record linked to the CI that was modified.

To add a new baseline, click New in the action bar of the Baselines table. This will open a new baseline record that is automatically linked to the CI:

History Tab

This tab has information about when the CI was created and updated, who created and last updated it, and all changes made to fields. It also contains the CI Version, which can be set to auto-increment if certain fields in the CI record change.

Configuration Item Processing

This section describes the processing of Configuration Items.

Knowledge Articles

Knowledge articles related to a configuration item can be searched for and selected in the Relevant Knowledge Articles table by clicking on the lookup icon.

Ownership

Configuration Item ownership is defined as the user whose Login matches the User Login field in the Configuration Item record.

Record Processing

Configuration Items follow a lifecycle workflow from ordering through retirement. Configuration Items are created in the state of Installed by default - see the Workflow diagram below.

Configuration Items that are requested but not immediately available might be created in a state of On Order, moved to In Stock when the Configuration Item is received, Pending Install while Operations is tasked with installing the Configuration Item, and finally Installed. Installed Configuration Items can change to In Maintenance for repairs and Retired or Stolen when the Configuration Item is no longer in use.

Workflow
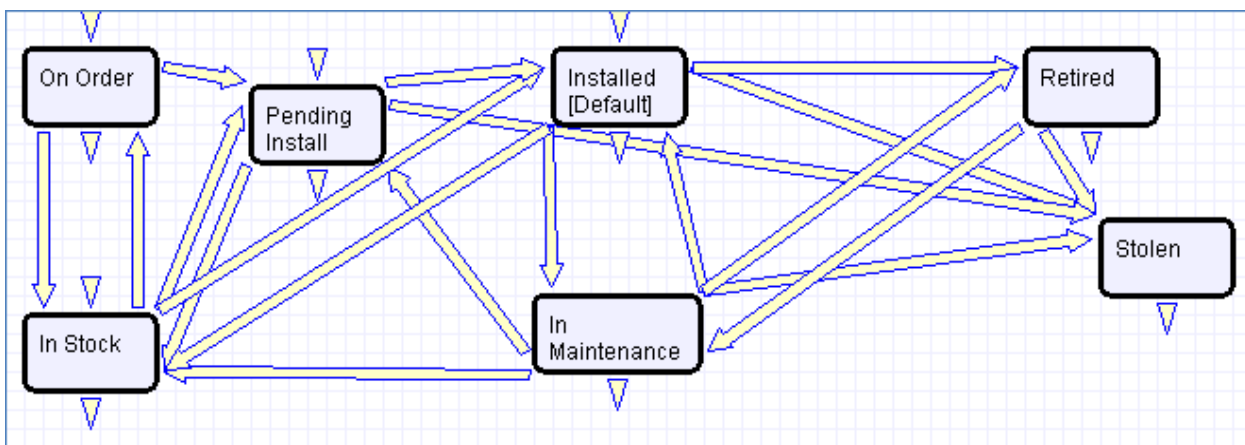
Integration with CMS and CMDBs

All Service Requests, Change Requests, Incidents, and Problems are fully integrated with the Configuration Management Database (CMDB). Every Configuration Item displays a related table for each request type, showing all of the Requests that the Configuration Item was included in. With the appropriate permissions, a user can easily open the Configuration Item from any of the Requests related to it, and view other related Requests to help identify, investigate, diagnose, and eliminate problems/issues more easily.

Each Configuration Item is also linked to a parent Catalog Item, which serves as a container record for the Configuration Item, and displaying all of the other Configuration Items for the Catalog Item in the Configuration Management System (CMS).

Automation

Configuration Items automation rules are described below:

Edit: All CI Edit Actions for status changes and updates (web or api)

This rule interacts with event management to send appropriate notifications when an outage or outage resolution event occurs.

An outage event sets the Operational status to Unplanned Outage. This rule triggers an immediate browser popup and sends an email to the CI Responsible Team and the System Admin Team about the outage. If the CI's Criticality was High or Critical, it also sends an sms message to the team's notification numbers.

If an outage resolution event occurs, the event updates the status from Unplanned Outage to Working, and this rule will then perform the same actions as for the outage, but with a resolution message telling people the outage has been resolved.

Rule: Edit: Create CI Changes (web or API)

Saved search: CI tracked attribute changed in last modification

This rule runs a script that generates one CI Change Log record for each change to a tracked field to show the old value, the new value, and the date/time/person who made the change. It also increments the CI Version field by .1 each time one of the tracked field is changed.

The tracked attributes by default are:

- CI Name
- CPU
- Hostname
- IP Address
- Memory
- Number of Licenses
- Operating System
- Ownership
- Version

These attributes are set in a global variable and can be changed at any time to include newly created attributes. The other details needed by the script are also set in global variables and can be changed as needed without having to modify the script.

TB: Delete Retired CI's a year after last update

Deletes all records that are Retired and last updated over a year ago. This rule runs once a month on CIs with a Status of Retired that were last updated over a year ago.

Charts and Reports

The Configuration Items table has the following pre-built charts and reports:

- All Configuration Items by Status, sorted by count
- CI's purchased in the past 12 months with cost
- CIs with Change Logs in last 90 days
- CIs with change requests in last 90 days
- Configuration Items by CI Subtype
- Configuration Items by CI Type
- Maintenance Report -CI's With Problems or Maintenance Time
- Number of In Stock CIs by CI Type then subtype
- Number of Installed CIs by CI Type then subtype
- Service Due within next 3 Months

Several of these reports and described in more detail below.

Report Name	Description
CIs Purchased in the Past 12 Months with Cost	This report shows the sum of the purchase cost for CIs per month purchased, for all CIs purchased in the past year. It has a both a graphical chart and a report component.
Number of In Stock CIs by CI Type and then Subtype	This report outputs information about the number of CIs that are In Stock, grouping them by CI Type and Subtype. It has a chart component and a report component.
CIs with Change Requests in last 90 days	This report provides a list of all CIs that were involved in change requests completed in the last 90 days.
Maintenance Report - CIs With Problems or Maintenance Time	This report shows the CIs that have been associated with at least one problem record and which have spent more than 1 hour in maintenance.
CIs with Change Logs in last 90 days	This report displays CIs that have been updated in the past 90 days. This is useful when integrating with an asset polling system to see all CIs that were modified through the integration recently.

Catalog Items Table

This table controls the basic configuration item information that may be used within an individual configuration item. It holds records that define details of the CI Type, CI Subtype, Vendor, Manufacturer, specific model numbers, pricing, and so on. Configuration Items that are based on catalog items link in the item on which the CI is based to prepopulate all of that information for a single CI.

Use Case

Catalog Items can be created and edited by members of the admin, admin import, and configuration manager groups. It is expected that the Configuration Managers will be responsible for maintaining the catalog information.

The form contains the following fields:

The list of CI Subtypes is held in a separate CI Subtypes table and is filtered once the CI Type is selected to only the subtypes associated with that type.   The Latest Unit Price gives an idea of the latest pricing for the item. Rules could be set up to push the price paid when a service request purchase is completed into this record to keep it updated as purchases are made.

Configuration Items Use of Catalog Items

Catalog Items are pulled into CI's, as shown below. When setting up a new configuration item, the user selects the CI Type, and then the CI Subtype, and then chooses a matching Catalog Item Name/Number from the available items. The items are filtered based on CI Type and CI Subtype, and once selected, their details are pulled into the CI record.

Automation

There is no automation on the Catalog Items table. It is purely a background table.

Items Requested Table

The Items Requested table acts as an intermediary table for the Service Request and Catalog Item tables. When users request items as part of a Service Request, they are creating new Item Requested records. These records represent the Items selected and their quantity for a particular request.

Record Creation

From a Service Request record, a user creates new Item Requested records for each Item desired in the embedded table view. After selecting the business service and service, in the Service Request record, clicking "Select Item" in the Items requested table presents the user with the new Item Requested form, which will be filtered based on the selected service to CI Types and Subtypes that were defined in the service.

The customer sees a very basic form to select the item desired. Staff members fulfilling the request see further information. They may look up Configuration Items in stock to see if they can fulfill the request without ordering additional equipment.

The Choose an In Stock Configuration Item section allows the user or fulfilling staffer to select an existing Configuration Item from the CI table. The CIs available for selection are filtered to only in-stock Configuration Items that match the CI Type and CI Subtype for this Item Requested. The CI fields link the Item Requested to a physical, tracked asset (CI) rather than just the template description (Item), enabling greater control and accurate record keeping. If they find a matching item, they see a button to assign that item to the user who requested it. If not, they can order the item and track its progress in the Status field.

Record Processing

When it is necessary to order new items, the responsible person changes the status to Ordered, which sets the Date Ordered field. Once the equipment is received, the status should be changed to Received, and at that point, a button becomes available to convert the item requested into a new configuration item.

Create Configuration Item

Clicking the button brings up the new configuration item screen, with the details already mapped for the CI Type, CI Subtype, Catalog Item, and so on. The rest of the information can be filled out and saved.

Later, once the configuration item is set to a status of Installed, a rule on the CI table comes back to the source Item Requested and updates its Status to Installed. Once completed, Item Requested records remain unchanged and stay linked to the Service Request. Since the Item Requested records are maintained separately from the Catalog Item records they were spawned from, they retain data that might change over time, such as Unit Price. Thus, even when the Catalog Items change, the Service Requests will always show Item data as it was at the time of the request.

Ownership

Item Requested records are "owned" by the Contact record whose Login matches the Creator Login field in the Item Requested. Since Items Requested are not maintained outside of a Service Request, the relevant ownership stems from the Service Request record.

CI Relationships Table

This table holds a record for each relationship between CIs. These relationships are used to create a diagram of the overall system of configuration items.

CI Relationship Fields

Each record has information about the type of relationship and the CIs involved. A new CI relationship has a Status of Active by default, but this can be set to Inactive if the relationship is no longer valid.

The CI to be related to other CIs can be chosen using the lookup next to CI Name. The section Select One or More Configuration Items allows you to choose the CIs that are related to the one selected above. If you would like to filter the CIs available for selection, use the Filter by CI Type dropdown and/or the Filter by CI Subtype fields.

Diagram of Relationships

The CI Relationships table is used to construct the diagram of relationships between CIs. The diagram is just a field with a special data type and it defines the CI Relationship table and the two CI fields within it as the elements to be diagrammed. The diagram is defined to show up to 5 levels above and below the configuration item from which is is launched.

Clicking on any item in the diagram takes the user to a view of that configuration item, from where the diagram may be updated to show all relationships to that item.

Relationship Types Table

This table stores a record for each type of relationship that can exist between Configuration Items (CIs). Each relationship type record also stores the Main CI Role, which keeps track of whether the relationship is started by a parent or child role, which in turn determines how the diagram showing asset relationships is structured. Each relationship also keeps track of the reverse relationship in the Opposite Relationship Type field.

The system comes with several built-in relationship types. Admin users may add further types as needed:

Relationship Type ↑	Main CI Role	Opposite Relationship Type
Connected by	Child	Connects
Connects	Parent	Connected by
Contained by	Child	Contains
Contains	Parent	Contained by
Distributed by	Child	Distributes
Distributes	Parent	Distributed by
Has registered	Child	Registered on
Impacted by	Parent	Impacts
Impacts	Child	Impacted by
Is Required By	Child	Requires
Registered on	Parent	Has registered
Requires	Parent	Is Required By
Used by	Child	Uses
Uses	Parent	Used by

Baselines Table

In the Baselines table, all information for a baseline is stored on the Details tab. Each baseline must have a name. A button can be used to restore the baseline to the associated CI. More information about the baseline can be entered in the Description field. The Date Baseline Created field is set by default to the date/time the baseline record was created.

The Configuration Item Details section shows information about the CI associated with the baseline. If the baseline is related to a release, the release can be chosen in the Release Information section. Any files and code can be stored in the Baseline Configuration Information section.

Event Management

Agiloft provides a robust Event Management system that can integrate with external monitoring systems or scripts, parsed inbound email, and other modules within the system to generate Events, convert specific types of event into incidents, and track information or warning events related to configuration items or other modules.

The system consists of two primary tables:

Event Types

This table is used to set up the event types that will be used to classify and process incoming events. The event types should be integrated with any external monitoring systems so that when that monitoring system creates an event, it also identifies the Event Type in our system (typically by ID) so that the event will be processed as predefined for that event type.

Configuration Managers and admin users have the permission to set up and modify event types. These permissions can be granted to other groups as needed.

Events

The Events table holds each individual event. Both events and event types have an Alert Type that defines the nature of the event and how the event is processed.

Events are most commonly related to Configuration Items and used to notify about problematic conditions for a particular CI. But they may be related to other modules or conditions in the system, such as SLAs that have shown a pattern of breaching too often, or OLAs that are not being met consistently.

Events Table

This table holds a record for each event that is created to indicate some piece of information or problem about a component of the system.

Out of the box, we have set up Events to be related to two main objects, shown in the Related To field: Configuration Items and SLAs. Depending on which relationship is selected, the corresponding event will be linked to a Configuration item or an SLA when the event is created.

If you want to relate Events to other objects, you can add new values to the Related To field and a new linked field set pulled from the other table in the system.

Event Creation

Events can be created in the following ways:

1. From an external monitoring system that polls hardware or software, identifies issues, and sends the information to Agiloft, either in a REST call, using the SOAP API, or in a pre-formatted email sent to an inbound account.
2. From a rule running on another table of the system that captures conditions that should provoke an event. As an example, we have a rule running on the SLA table on a weekly basis that finds SLAs whose current percentage of breached incidents is > 75% and creates an Event for each such SLA. Similar rules can be created on any table of the system and can result in an event being created. That event will then generate an incident or not, based on the setup of the event type.
3. From another external system that runs standard processes that should result in an informational event. These can be created in the same way as for a monitoring system: using REST, SOAP, or parsed email.
4. Via scheduled import from another system. Scheduled imports can run once a day or more frequently to pull in data from other systems. This import file could create events.
5. Events can also be created manually by a user with the permission to do so (admin and configuration manager groups).

Event Layout

The event form has information pulled from the event type as well as case-specific details provided by the monitoring system.

Event Details Tab

The Event Details tab has most of the relevant fields. Below is a typical outage event:

Event

Save ▾ | Cancel ▾ | Event Details | Related Events | Communications | History | « »

ID	47
Event Summary	Cannot login to production firewall server1
Alert Type	Outage ▾
*Event Type Summary	Server Login Failure ▾
Status	Incident Created ▾
*Event Type Description	Failed Login to Server x times
*Related To	Configuration Item
People to Email	
*Investigation Procedures	Check if https://serverurl/gui2 is accessible. Try to login to demo/systemmonitoring/other KBs. If it's not accessible and login doesn't work SSH to server and

take thread dump "XXHOME/bin/dump_for_debug -r" and review it for hanging/heavily loaded threads

check tail -f -n 100 XXHOME/logs/console.log for memory problems (you may need to take a heap dump to review it later)

check mysql's "show full processlist\G" for running DDL/locks |
| Update Notes | |
| Event Details | Firewall server seems to be down. It cannot be accessed. |

The Event Summary and Event Details are provided by the monitoring system to describe this specific event, while the Event Type Summary and Event Type Description and People to Email come from the related Event Type. The continuation of the Event Details screen shows the related configuration item or SLA and also a link to any incident that was created.

Related Events Tab

The Related Events tab shows any preceding event or duplicate events as well as any precipitating event, for a resolution event only.

Communications Tab

The Communications tab shows any emails sent by the event automation.

History Tab

The History tab shows the date and time the event was created and by which monitoring system or person. It also shows the time created (for reporting) and tracks any changes to the event record.

Incident Creation

Serious Warning events and Outage events should almost always result in an incident, so that standard procedures can be followed and SLAs invoked.

When an event is created for an event type that should create an incident, a rule runs to create the incident, mapping the Event Summary and Details, urgency and impact based on the values in the event type, mapping the configuration item, and setting its Business Service to IT Service Issues, the Service Problem Type to Event Monitoring Incident, and the Incident Reported Via to Event Monitoring System. The incident resulting from the above event is shown below:

The incident will then be handled like any other incident.=

Updating the Configuration Item

In addition to creating an incident, if the Alert type of an event is Outage and there is a linked Configuration Item, the system will update that Configuration Item to set its Operation Status to Unplanned Outage. It will also add a note to the Status Notes and Updates field identifying the event.

Resolution Events

When an event of type Outage Resolution occurs, the actions taken by the outage event are effectively reversed. In the example shown below, a self-resolution took place and this event was followed by an Outage Resolution event:

ID	49
Event Summary	Firewall server is working now
Alert Type	Outage Resolution ▾
*Event Type Summary	Server Login Failure ▾
Status	Logged ▾
*Event Type Description	Login to server is successful after a reported failure
*Related To	Configuration Item
People to Email	● Jack Wicks 🔍
*Investigation Procedures	
Update Notes	
Event Details	The server can now be accessed and is no longer down.

The Related Events tab shows any linked events, both duplicate events that followed this event, and the precipitating event for a resolution event:

Save ▾	Cancel ▾	Event Details	Related Events	Communications

ID	49

Precipitating Event Information

Precipitating Event ID	47 🔍	Precipitating Event Incident ID	288
Precipitating Incident Status	Closed		

Duplicate Events

Previous Event ID	🔍
Duplicates	Status: No records.

In this case we see that event 49 resolved the problem reported by event 47. As a result, the precipitating event's Incident was updated to Closed and the Configuration Item Operational Status, which had been set to Unplanned Outage with the first event, was set back to Working by the second event. Each time an event updates an incident or a configuration item, it adds notes about the event ID and the actions it is taking to that record.

Event Processing and Business Automation

For information and warning events, or events that do not generate an incident, if the event type is defined to send email, then emails are sent to the appropriate team or people notifying them about the event.

When the event is created, the Status is also set automatically based on the following rule:

Create: All Event Creation Actions - update CI, generate incident and send emails

- U: Set Duplicate End Date to Date Created plus duplicate minutes
    - This rule sets the Duplicate End date field in the new event by adding the event type's duplicate minutes value to the date created. This will be used to compare to any new events of the same type that are created (if their date is less than that date, they will be marked as duplicates).
- I: All Event Create Actions
    - The Previous Event ID field is set automatically (based on the field default value action) to any event whose SLA ID or CI ID matches the new event, its related to matches, and its event type matches, and its date is the latest one that is earlier than the new event. So if there is a previous event matching that criteria, the Status of the new event is set to Duplicate:
    - Otherwise, if it is an event type that should create an incident, the incident is created and the Status is changed to Incident Created.
    - Otherwise, if emails should be sent, then the notifications are sent based on the selection in the Event Type:

This rule also handles the events that are outages or outage resolutions, and it also links the warning resolution to its precipitating event, so if the warning event generated an incident, the incident can be closed.

A second creation rule checks whether the precipitating event field has a value, and if it does and an incident was generated, it closes the incident with an appropriate status note update:

Create 2: If resolution event, update linked Incident
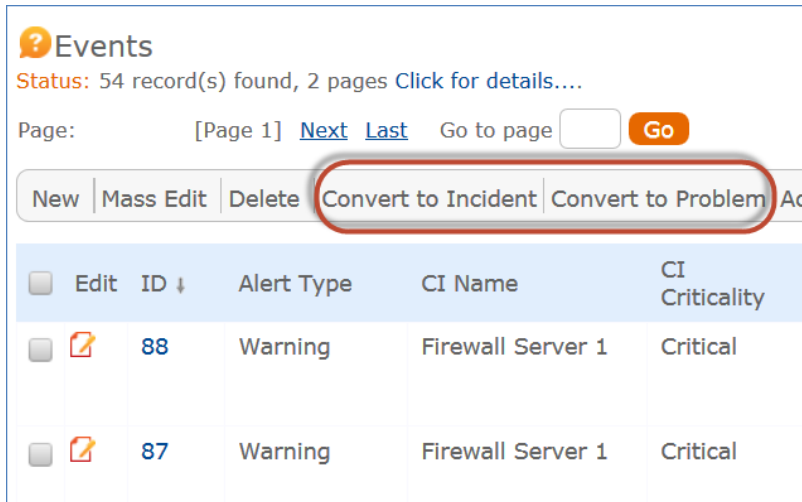
Search: Alert type is warning resolution or outage resolution and linked open incident exists This runs a linked action to update the linked incident (if it is not already closed) with the details from the event and to set its Status to Closed and its Closure Category to Self-Repaired by Monitoring System.

Updating the Event Itself

While most events that require followup are expected to result in an incident, it is also possible to update the event itself. Service desk, Service managers, and Configuration managers can all update the event record as needed to provide investigation notes or other information. This information goes into the Update Notes field.

Followup - Converting to an Incident or Problem

It is also possible to convert from an Event to an Incident (for those that did not automatically generate an incident) and to a Problem, in case the number of warnings seems to suggest a problem. Buttons are available on the action bar for this conversion:
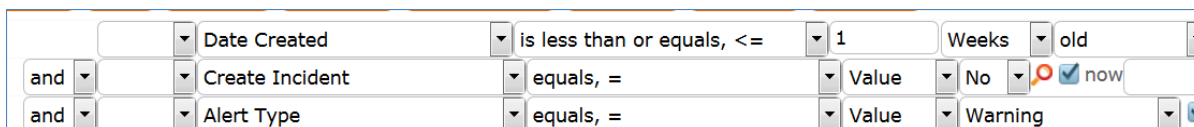
Summary Conditions Business Rules

It is possible to set up summary conditions that look for a certain number of events of a certain type for a specific CI or SLA and notify if that condition occurs. The system contains one example of this kind of rule as a model that can be followed to create additional rules based on specific criteria.

	Edit	Rule ID ↓	Comment	Rule Type	Name of Action(s)	Priority
☐	✎	327	Summary rule: more than 4 warnings for a CI that don't generate incident in a one week period. Notify CI Responsible team	When some summary condition based on multiple Event has/hasn't been met.	Email Responsible team for more than 4 warnings in last week	12

This rule runs once a week early Monday morning and looks for non-incident-generating warning events created in the past week.

Then it groups by CI ID and looks to see if there are more than 4 such events for any Configuration item. If there, are, it collates the events into an email and sends it to the CI Responsible team for that CI.

Archiving and Deletion

Events can be archived or deleted based on any time criteria. We have defined a sample rule that will delete information events after 3 months:

TB: Delete Information Events after 3 months

This rule runs weekly and finds events whose Alert Type is information and that were created more than 3 months ago. It runs a deletion action to delete the events.

Reports

There are several reports set up on the Events table. These can be used to review the areas where problems are occurring, to identify problem configuration items that consistently generate events, and to track the effectiveness of the event monitoring system. Below are some samples of some of the built-in reports. These reports typically also provide an html and/or excel-based output with more details.

All Warning Events in past month, by CI
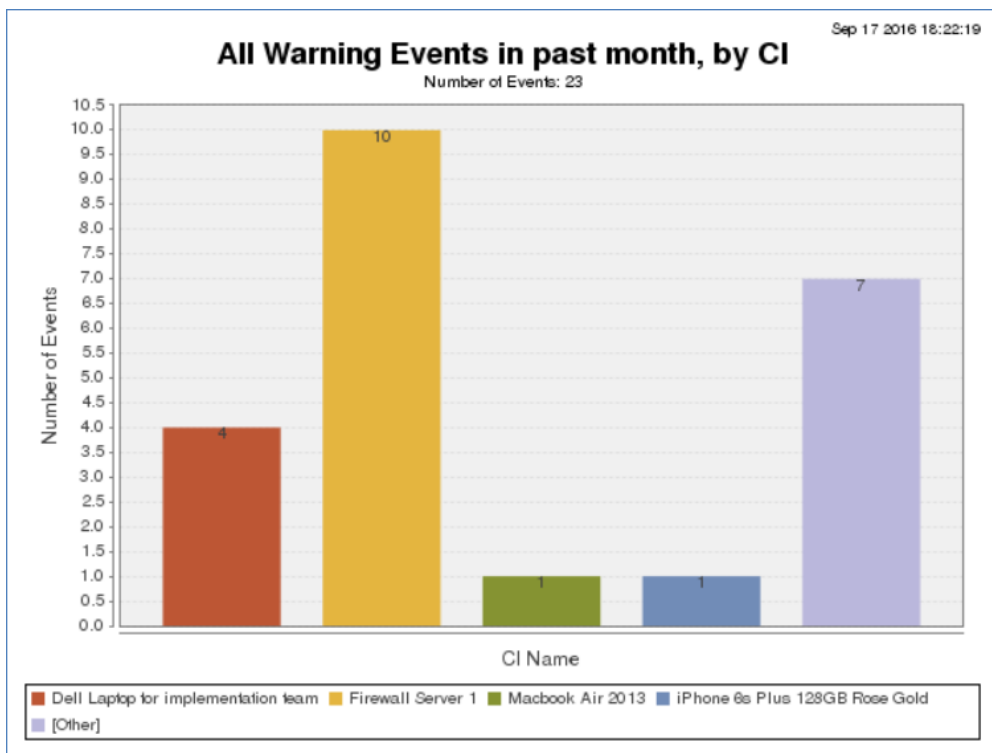
Duplicate Events by Event Type

This chart shows all the duplicate events created, totaled by event type:

Duplicate Events by Event type
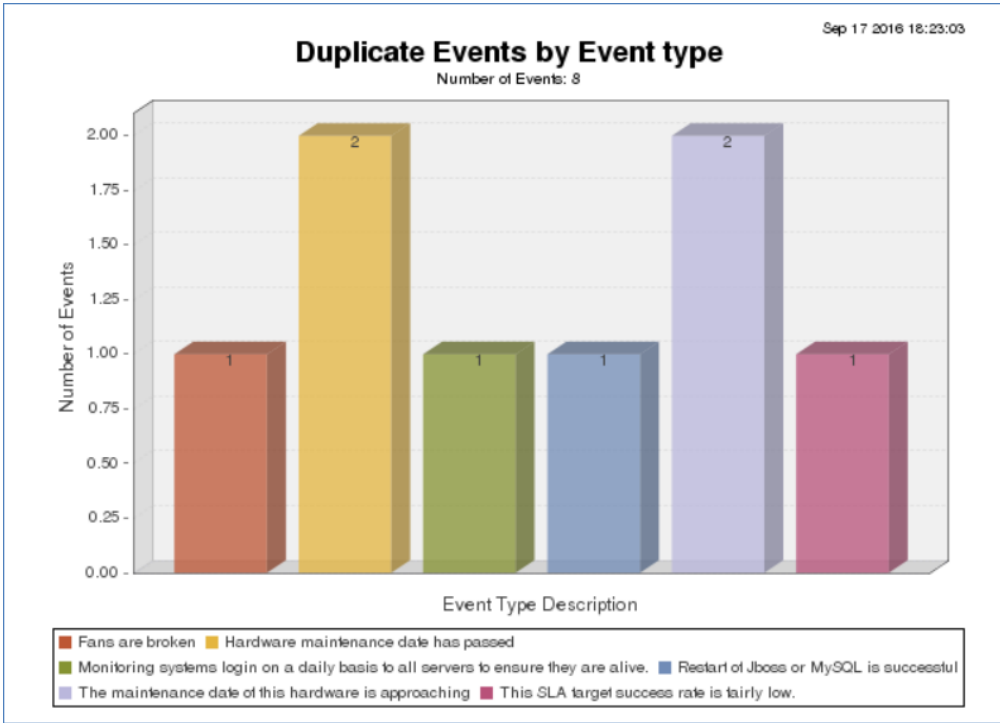
Events Created by Time of Day for the past week

This groups events by the time of day they were created to show peaks and valleys in event activity.

Events Created by Time of Day for the past week
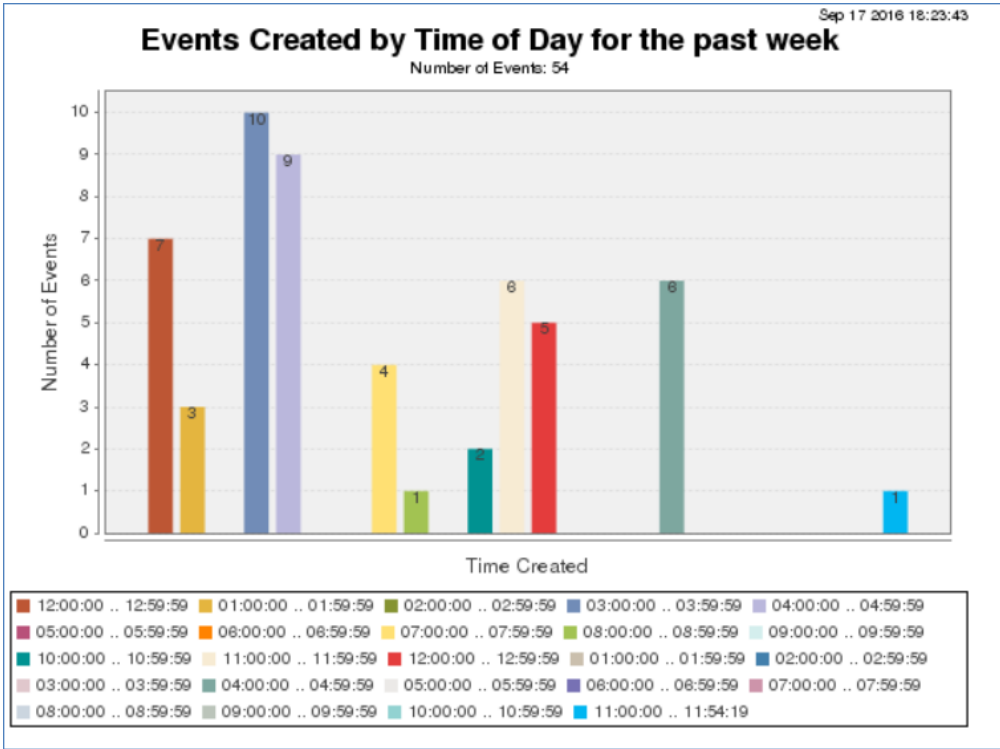
Events Resulting in Incidents by Event Type

This shows just the event types that result in incidents.
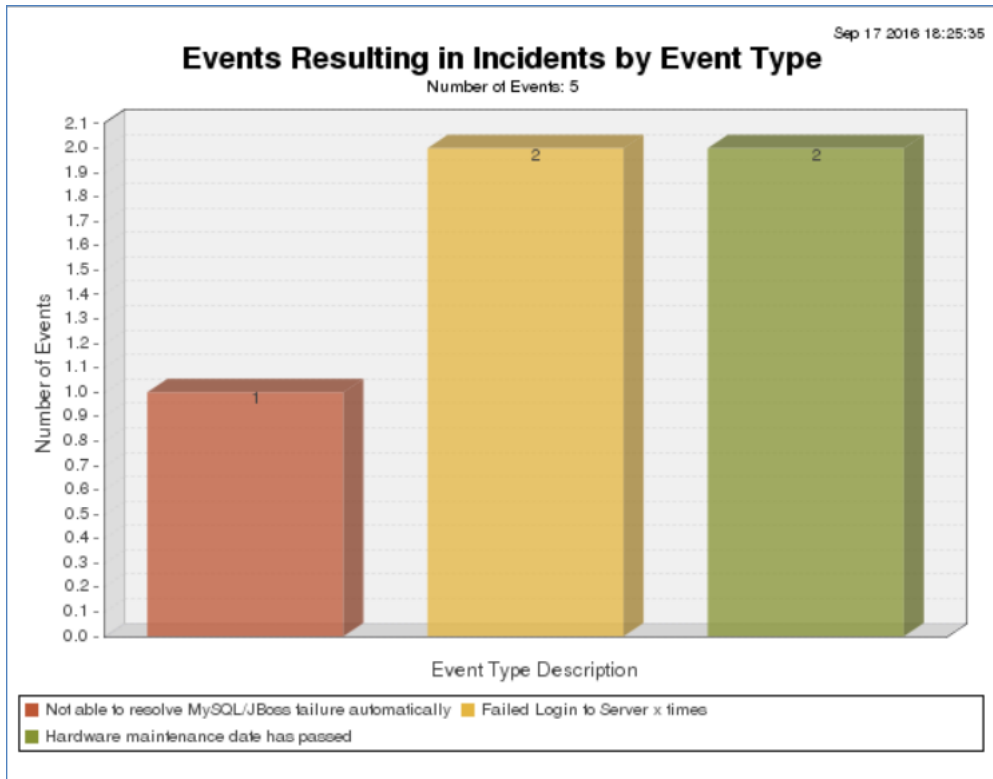
Non-Duplicate Events per CI Name

This report outputs both an html and excel report showing the non-duplicate events grouped by CI:

Non-Duplicate Events per CI Name

Number of Events : 23

CI Name : Brother 6180 Printer 3

    Number of Events : 1

ID	Alert Type	CI Name	CI Criticality	Event Type Summary	Event Details
28	Information	Brother 6180 Printer 3	Medium	Hardware Maintenance Date Approaching	Printer is flashing, requires maintenar

CI Name : Dell Laptop for implementation team

    Number of Events : 3

ID	Alert Type	CI Name	CI Criticality	Event Type Summary	Event Details
16	Warning	Dell Laptop for implementation team	Medium	Laptop Overheating	
17	Warning	Dell Laptop for implementation team	Medium	Laptop Overheating	
20	Warning	Dell Laptop for implementation team	Medium	Laptop Overheating	

CI Name : Firewall Server 1

    Number of Events : 12

Event Types

The Event Types table holds all of the possible event types, categorized by the Alert Type and whether they pertain to a configuration item or some other object in the system. Out of the box, the system has event type relationships with Configuration Items and SLAs, but other relationships may easily be added.

Only members of the admin and Configuration Manager groups can create and modify the event types. For use with automated monitoring systems or scripts, it is important to create appropriate event types and then update the monitoring system to push the ID for the correct event type into the events it creates. This way, the processing details defined in the event type will be used to determine the correct course of action for the event.

The system comes with several sample event types showing some of the different options for processing events:

Alert Type	Event Summary ↑	Event Type Description	Email Behavior	Create Incident
Outage	AWS Status Check Failed	Alarm for Status Check Failed has been triggered by AWS=>SNS=>Lambda	Do not Send Email	Yes
Information	Backup Service Started	Backup service started, estimated duration is 2 hours.	Send Email	No
Information	Daily login to check system health	Monitoring systems login on a daily basis to all servers to ensure they are alive.	Do not Send Email	No
Warning Resolution	Disk Space Emergency	Disk Space Emergency is resolved and disk space is now adequate	Do not Send Email	No
Warning	Disk Space Emergency	disk space is extremely low	Do not Send Email	Yes
Information	Disk Space Emergency	Disk Space Emergency is resolved and disk space is now adequate	Send Email	No
Information	Hardware Maintenance Date Approaching	The maintenance date of this hardware is approaching	Send Email	No
Warning Resolution	Hardware Requires Maintenance	Hardware has received its maintenance	Send Email	No
Warning	Hardware Requires Maintenance	Hardware maintenance date has passed	Do not Send Email	Yes

There are five different Alert Types, and each may be handled differently:

- Information
- Warning
- Outage
- Outage Resolution
- Warning Resolution

When event types are set up, the Outage Resolution and Warning Resolution alert type events should be linked to their preceding precipitating event (the warning or outage events). In this way, when a resolution event is created, it can update any incident or configuration item that was linked to the precipitating event to close the incident, change the status of the CI, and so on.

Event Type Creation

Here is a typical setup for an outage event type:

ID	8	Alert Type	Outage ▾	Related To	Configuration Item ▾

Event Type Details

*Event Summary	MySQL/Jboss Failure/Resolution
Event Type Description	Not able to resolve MySQL/JBoss failure automatically
Investigation Procedures and Tips	Troubleshooting steps: 1. Check whether the jBoss process is shown as started 2. Check whether MySQL is running Edit

Processing Information

Time Range for Duplicates	30	Email Behavior	Do not Send Email ▾
Create Incident	● Yes ○ No		
*Incident Impact	Department ▾	*Incident Urgency	High ▾

Note that you must provide an Event Summary for the event type. This field is not required to be unique. The Investigation Procedures and Tips field is an HTML-formatted field so you can include screenshots or formatted steps for investigation. This field will be shown in the resulting event.

If several events of the same type for the same configuration item are received in short succession, it is important to be able to identify the later ones as duplicates, so we do not create incidents from each one. When setting up the event type, the Time Range for Duplicates field is used to provide a time range in number of minutes, after which a new event of the same type and for the same CI should NOT be considered a duplicate. In the case above, after 30 minutes, if a new event of this type is received, it will be treated as a new event, while before that, it will be marked as a duplicate.

You can choose whether the event should send an email to a team or to individuals. You can also choose whether the event should create an incident. When creating an incident, we typically do not send an email from the event, since the incident creation rules will already send the necessary emails. If the event is going to create an incident, you must set the default impact and urgency to use for the incident, so that the priority and SLA targets are all set.

Resolution Events

When creating an event type whose alert type is a resolution, there are some additional fields to fill out, so that the resolution event type is linked to the precipitating event type. Here is the resolution event type tied to the outage event type shown above:

ID	11	Alert Type	Outage Resolution ▾	Related To	Configuration Item ▾

Event Type Details

*Event Summary	MySQL/Jboss Failure/Resolution
Event Type Description	Restart of Jboss or MySQL is successful
Investigation Procedures and Tips	

Edit

Processing Information

Time Range for Duplicates	30	Email Behavior	Do not Send Email ▾
Create Incident	◯ Yes ⦿ No		

Precipitating Event

*Precipitating Event Summary	MySQL/Jboss Failure/Resolution	*Precipitating Event Type ID	8
*Precipitating Event Description	Not able to resolve MySQL/JBoss failure automatically		

By selecting the related event type here, you create a link that can be used in the resulting event to pull in the precipitating event and its Incident ID and Configuration ID, which will allow the system to update those related records when a system self-corrects.

Email Notification Options

For events that don't create an incident, you have some additional choices for email notifications that can be sent from the event itself. When you choose Send Email in the Email Behavior field, two new fields appear. You can choose whether to notify the CI Responsible team about the event (for a Configuration Item Event) or the selected people in the People to Notify field, or both. When the event is created, your selection here is used to determine who should receive an email about the event.

It is of course possible to have event types that do not create an incident or send an email, but that are just informational. For more information on how the event types are used, see Events Table.

Release Management

The goal of release management is to allow planning and deployment of releases into live environments. A release can be linked to one or more change requests or service requests and those requests and their tasks will be visible within the release record.

Releases are based on Release Types, which are preconfigured with documentation and task workflows that may include the approvals necessary to proceed with the release.

Creating a New Release

The main Details tab of the layout has the main fields that identify the owners of the release:

When the Release Type and Version is selected, this pulls in the Release Readiness Criteria and Acceptance Criteria and Checklist documents that are attached to the release type, and these documents provide the details for how the release should proceed, what the go / no go criteria are, and so on. Fields for Build Manager, Change Manager, and Test Manager identify the main decision makers and drivers of the release process.

CIs and Change Requests Tab

The CIs and Change Requests Tab is used to select the Configuration Items for the release. From here you will be able to view the graphical display of the CI's to see any dependencies or downstream CI's that will be affected by the release.

If any definitive media is involved, those records can be looked up and pulled into the Definitive Media table.

In addition, you can search for any outstanding change requests and link them to the release. The change request can also be linked to the release record from within the change request:

Schedule and Tasks Tab

The release start and end window is defined on this tab. In addition, any task workflow is shown here and tasks are generated and launched from here:

Change Request Tasks

If change requests have been linked in and they also have tasks, the change request tasks will be shown below the release tasks:

Related Records Tab

This tab shows any related Knowledge articles, incidents, and service requests related to the release. New service requests for equipment or license purchases can be generated directly from within the release in order to be sure the proper goods are provisioned prior to doing the release.

Releases

The Releases table holds a record for each release, with information about the release plan, responsible people, notes, and tasks. Releases can be linked to configuration items, incidents, change requests, service requests definitive media, and knowledge articles.

Working with Releases

New releases are created by going to the Releases table and clicking New on the action bar.

Release Layout

The tabs of the Releases table are described below:

Common Area

The common area of releases contains the Release Summary, which is a brief description of the release's purpose. The Status field provides an overview of the release's progress. Status options are Planned, Scheduled, Build in Progress, In Testing, Deploying, Post-Deployment QA, Completed, and Rolled Back. The default value for Status is Planned.

Release Package Type has options of Major Release, Minor Release, Significant Release, and Emergency. Release Type and Version allows this release to be linked to the appropriate Release Type, stored in another table. The combination of Release Package Type and Release Type defines the workflow tasks and approvals that will be needed for the release.

ID	26		Status	Planned
*Release Summary	Upgrade to Symantec Commander v. 12			
*Release Package Type	Minor Release		*Release Type and Version	Software application upgrade V. 1

Details Tab

The Details tab stores the core information about releases. Release Description allows for a longer explanation of the release's purpose and a description of what will be included in the release. Release Version allows a number to be assigned to the release.

When the Release Type and Version is selected, this pulls in the Release Readiness Criteria and Acceptance Criteria and Checklist documents that are attached to the release type, and these documents provide the details for how the release should proceed, what the go / no go criteria are, and so on. A Build Manager, Change Manager, and Test Manager may be assigned to the release. The list of available Build Managers is filtered to employees on the Release Management Team, while the available Change Managers are all employees on the Change Management Team. Any employee can be selected as the Test Manager.

Roll-out Plan allows for files describing the plan for rolling out the release. Risk if Done and Risk if Not Done

have options of Low, Moderate, High, and Very High.

Risk Assessment is a text field allowing for a more detailed description of the risks involved with the release. The Backout Plan for Reverting to Baseline holds files about the steps involved in the release backout plan, and any configuration files needed to roll back the release.

Closure Category is filled out upon completion of the release to be used in reports. It is visible when the Status is Completed or Backed Out. The closure categories are the same choices available for change requests, and the closure category of the release could be pushed to the related change requests automatically. Working Release Notes is used for any notes throughout the release process. The author, date, and time of the note are captured.

The Testing Plans and Results file field is used to attach both the original test plans and any test results or data produced through the testing tasks.

CIs and Change Requests Tab

The CIs and Change Requests Tab is used to select the Configuration Items for the release. From here you will be able to view the graphical display of the CIs to see any dependencies or downstream CIs that will be affected or used in the release. The Configuration item view shows the location of the CIs as well as their criticality.

If any definitive media is involved, those records can be looked up and pulled into the Definitive Media table to identify the licenses and/or software involved in the release. In addition, you can search for any outstanding change requests and link them to the release. The change request can also be linked to the release record from within the change request.

The lookup icon on the action bar of each table can be used to associate one or more configuration items, definitive media items, or change requests with the release.

Configuration Items affected by this Release

Status: 1 record(s) found, 1 pages. Click here to re-count records.

New | Mass Edit | Delete | 🔍 Unlink | Actions ▾   Views ▾ | Search ▾ | 🏠 ▾ | 🖨 ▾

	Edit	ID ↓	CI Name	CI Type	CI Subtype	Status	Criticality of CI	Ownership	Location Name	CI Role and Description
☐	📝	36	Main Exchange Mail Server	Hardware	Server	Installed	Medium	IT Item not shown to Users	Data Center	

Definitive Media

Status: No records.

New | Mass Edit | Delete | 🔍 Unlink | Actions ▾ | Views ▾ | Search ▾ | 🏠 ▾ | 🔍 ▾ | 🖨 ▾

Change Requests implemented as part of this Release

Status: 1 record(s) found, 1 pages. Click here to re-count records.

New | 🔍 Unlink | Views ▾ | 🖨 ▾

	Edit	ID ↓	Status	Change Summary	Change Type	Risk If Done	Business Service	Service	Change Window Start Time	Change Window End Time	Requires Downtime	Number of Linked CIs
☐	📝	280	Approved	Upgrade Email exchange server	Normal	Low	IT Infrastructure Change	Server Change involving downtime	Jul 11 2016 00:00	Jul 11 2016 02:00	Yes	3

Schedule and Tasks Tab

This tab contains date fields and action buttons to manage the time frame of the release. The Release Window Start Date and Release Window End Date can be chosen manually. The Earliest CR Start Time and Latest CR End Date default to the corresponding value in any linked change requests for this release where the Change Window Start Time is less than all other change requests and greater than all other change requests, respectively.

In addition, any task workflow is shown and tasks are generated and launched from here. The workflow and activities for a release are all handled within the tasks, which are pre-configured for each Release Type and Release Package Type. The workflow is defined in the Release Type record, and the workflow tasks may include conditional tasks and/or approvals based on the Release Package Type.

Once the tasks are generated and launched, if any task is marked as Failed, that will send an email to the three managers identified in the release and prevent the workflow from progressing. Once the Generate Tasks button is pressed, the tasks for the selected workflow are automatically generated, and placed in a Queued status. The tasks may include approvals as part of the workflow, in addition to tasks that require specific work:

ID	Sequence ↑	Task Type	Task Summary	Due Date ✎	Status ✎	Assigned Team ✎
1565	1	Task	Plan and assign available and relevant developers to Release Build		Queued	Project Manager Team
1566	2	Task	Conduct specification finalization meeting with relevant developers and Build Manager		Queued	Custom Applications Team
1572	3	Task	Create relevant release documentation		Queued	Document Creator Team
1567	3	Task	Build out custom code needed for application upgrade		Queued	Custom Applications Team
1568	4	Task	Pre-deployment testing in development environment by developers		Queued	Custom Applications Team
1569	5	Task	Merge code into a release and pass off to Quality Assurance team for signoff		Queued	QA Team
1571	6	Approval	Final Review and Approval by Test Manager		Queued	QA Team
1570	6	Approval	Review and Approval by Build Manager		Queued	Custom Applications Team
1573	7	Approval	Compliance review and approval of Release and Documentation		Queued	Compliance Team
1574	8	Task	Deploy Release of application upgrade		Queued	Release Management Team
1575	9	Task	Conduct post-deployment QA and smoke testing		Queued	QA Team
1576	10	Task	Conduct Change Evaluation and Service Validation and Acceptance		Queued	Change Management Team

Launching the tasks will set all tasks that have no prerequisites to Assigned and notify the team/person that the task is due. More on the task process can be found in the Task Management section, and a description of the tasks table in particular is in the Tasks Table section.

Adding Calendar Events

Once the Status has been changed to Scheduled, calendar events can be generated using buttons on the screen. The Release start and end window event can be created by pressing the Add Window to Calendar button. That will create an event of type Implementation. Clicking the Add Another Event to Calendar will bring up the Calendar record to be filled out and will link that record back to the release. Any calendar events created will be shown below:

Release Schedule

Release Window Start Date	Jul 07 2016	00:00		Release Window End Date	Jul 08 2016	00:00

Earliest CR Start Time   Jul 07 2016 00:00   Latest CR End Time   Jul 08 2016 00:00

[Add Window to Calendar]   [Add Another Event to Calendar]

Calendar Events

Status: Calendar Event #49 successfully updated. 2 record(s) found, 1 pages. Click here to re-count records.

☐	Edit	ID ↓	Start Time	End Time	Duration	Event Type	Description	Linked Records	Requires Downtime
☐	✎	49	Jul 06 2016 08:00	Jul 06 2016 10:30	0 days 02 hours 30 minutes	Build	Build the sample setup - Implement REL-ID:26	REL_26	No
☐	✎	48	Jul 07 2016 00:00	Jul 08 2016 00:00	1 days 00 hours 00 minutes	Implement	Upgrade to Symantec Commander v. 12 - Implement REL-ID:26	REL_26	No

The calendar can be subscribed to as an internet calendar and shown in your company's calendar software, whether that is Outlook, Exchange, or Google, for instance.

Change Request Tasks

If change requests have been linked in and they also have tasks, the change request tasks will be shown below the release tasks/

Related Records Tab

Knowledge articles applicable to the release are included on this tab. In addition, incidents and service requests related to the release can be selected here. The New button on the action bar of the related table can be used to create a new knowledge article, incident, or service request related to this release.

New service requests for equipment or license purchases needed for the release can be generated directly from within the release in order to be sure the proper goods are provisioned prior to doing the release. The costs from any purchases related to the release can be accessed through any linked service requests. Existing records can be looked up and linked in using the lookup icon above each table.

History Tab

The History tab contains the auto-generated Date Created and Date Updated, and the creator and last updater are stored. Below, the History field shows a row for each change made to the release record, including the updater, date and time of the update, the field(s) updated, and the old and new values of the field(s).

Completing the Release - Updating CIs

When the release has been completed and all tasks are done, the Status and the Closure Category fields should be updated to indicate whether the release was completed successfully.

In addition, if CIs were updated in such a way that new relationships should be created for them or changes are made to their attributes, then those CIs should be updated. This can be done in the simplest cases by using Mass Edit from the linked CIs.

Selecting CIs and clicking the Mass Edit button allows the user to choose which attributes of the linked CIs should be updated and to update them all at the same time. In addition, all CIs included in the Release will automatically show the release within their own Requests and Work Done tab. Any change requests tasks will also appear within the Configuration item record.

Release Types

Overview

This table stores a record for each type of release, each with a default task workflow and documentation for release readiness and acceptance criteria. Release Types are pulled into Release records - see Releases - where all of the workflow and processing information is pulled in and applied to the Release.

Release Type Overview

Release types can be marked as Active or Inactive by changing the Status field. Active is the default value, but Status can be set to Inactive if the release type will no longer be used.

The Clone to Create new Version button can be used to create a new release type with the Release Type, Description, Acceptance Criteria, Default Task Workflow, Release Readiness Criteria, and Task Generation Method mapped. The new release type record will appear on the screen so that any modifications can be made to these fields. The Version Number of the cloned release type is set to the previous release type version number incremented by 1.

Release Type and Version is a compound field, combining the Release Type name and the Version Number of the release type. The Release Package Type can also be incorporated into the Release Type to serve as a model for that Package Type - i.e, Major Release, Significant or Medium Release, Minor Release, Emergency Release.

Release Readiness Criteria and Acceptance Criteria and Checklist hold files describing release readiness and criteria for acceptance of the release - i.e. policies, regulations, standards, other requirements that must be met for deployment approval. These can also include information about Service Validation, various test plans and other testing activities.

In the tasks section, a Default Task Workflow Title can be selected. This determines the Task Workflow that is chosen automatically in a new release for this release type. These can be customized and used to incorporate Service Test Models, Service Validation, and various testing activities. The Task Templates are also used to manage roles and responsibilities and schedules, and can include different activities in different deployment environments. Once a Default Task Workflow Title is selected, the task templates included in the workflow are shown in the Task Templates table below.

Knowledge Management

The goal of IT Knowledge Management is to make information available and easy to maintain. Agiloft has several tables designed to maintain the service knowledge management system. These tables are Knowledge Articles, Knowledge Ratings, Knowledge Subtypes, and Knowledge Topics.

- Knowledge Articles

Knowledge Articles

Records in the Knowledge Article table represent FAQs, policies, user guides, and other reference information that is helpful in maintaining and improving the ITIL process.

A knowledge article record has fields that are divided into tabs: Details, Progress, Source, Emails, and History. A small number of fields are stored in the common area, which is visible at the top of all of the other tabs.

Use Case

Knowledge articles may be created by members of the admin, Admin Import, Base ServiceDesk, Business Admin, Document Creator, Document Manager, Document Reviewer, ITIL Business Admin, Marketing, and Project Manager groups.

Knowledge articles are generally created manually by clicking New within the table view or by converting a resolved incident, service request, problem, or change request into an article.

If a user attempts to create a knowledge article with the same title as a previous knowledge article, the system will prevent the new article from being created until it has a unique title:

Error: The Title field has the "unique value" attribute, and a record with the value of Confluence User Guide already exists.

Knowledge topics are assigned to knowledge articles based on content. Users can subscribe to topics in order to get emails when new articles with those topics are created.

Knowledge Article Fields

The Knowledge Article fields are listed below:

Common Area

The common area of each knowledge record shows the ID, which is unique within the Knowledge Article table and auto-assigned to each new record. In addition, the Status of the knowledge article is shown to provide information on the article's progress through the review and publishing stages. The default Status is Draft for new articles. Other Status options that can be used to track progress include Pending Document Manager, Pending Approval, Ready for Publication, Published, Retired, and Canceled.

The common area also contains an Assigned Team dropdown. The Assigned Team defaults to the Document Management Team, but may be updated to hold another team. If an individual is responsible for the article, his name can be selected in the Assigned Person dropdown. The options for Assigned Person will be filtered based on the Assigned Team chosen.

The common area also contains three buttons when the Status is Draft: Cancel this Document, Submit for Review, and Submit for Approval. Cancel this Document changes the Status to Canceled. Submit for Review changes the Status to Pending Document Manager and triggers an email to the Document Management Team. Submit for Approval changes the Status to Pending Approval and triggers creation of approval records for each individual who has been selected as one of the Approvers on the Progress tab.

Details Tab

The Details tab of a knowledge article contains most of the basic information about the article. There are five main sections: Document Details, Document Files, Related Articles, Ratings, and Submitter Information.

Document Details

- Knowledge Type is required and helps to categorize the article. Once a Knowledge Type is chosen, the available Knowledge Subtypes available will be the subtypes related to the Knowledge Type. Topic(s) should be selected based on the content of the article. If any users have subscribed to one of the topics chosen for this article, then they will receive an email when this article is published notifying them that there is a new piece of information they might find interesting.

- Priority and Rush can be used to indicate how quickly this article should be reviewed and published. The default of Rush is No.

- Purpose and Description are used to explain the content of the article. HTML can be entered into the Description field by changing the radio button below it to HTML, or by clicking the Edit button.

- Special Compliance Requirements is required, and one or more options can be selected. If any option besides None is chosen, Compliance Notes must be entered.

- Document Access is required and used to identify who should be able to view this article. The options are Public, Partners, All Internal Users, Restricted to Specific Departments, and Restricted to Specific Users. If Restricted to Specific Departments is selected, a field appears allowing selection of one or more departments to which this article should be available. Similarly, if Restricted to Specific Users is selected, a field will be visible to allow lookup and selection of those users. Audience is an informational field, providing an overview of the types of people for which this article is intended.

Document Files

This section has information about the document itself. Source Files contains the original files used for the knowledge article. Published Format has several file types to represent the final, published version of the document. One the article is published, the finalized document(s) can be added to the Published Files field. Both Source Files and Published Files accept document (Word, PDF, etc.) and image file types. If the file is stored at in internal URL, the URL can also be entered in the Internal File URL field.

Related Articles

This section has one table storing information about other articles similar to this one. New articles can be added to the Related Articles list by clicking the lookup

 icon . Below this table are two fields: Overall Rating and Ratings Average, which represent the average rating description and number for this article based on all ratings.

Submitter Information

This sections shows the Submitter's name and department. The submitter defaults to the person who created the knowledge article, but an admin or manager can edit the Submitter if the article was created on behalf of someone else.

Progress Tab

This tab holds approval information and notes on the knowledge article. Requires Approval is set to Yes if the article must be approved; if so, one or more Approver(s) can be selected. Any Approval Notes can be entered into the field manually. In addition, if any approvers write notes into an approval record, those notes are pushed to the Approval Notes here in the article.

Working Notes holds more general notes on the article's progress.

If there are milestones for the article, the Next Milestone Date can be stored. An article may also have a Final Due Date.

Services Tab

The Services tab contains a table of Associated Services. Services related to this article can be selected by clicking the lookup icon.

Source Tab

The Source field is used to specify where the knowledge article originated from a change request, incident, problem, or service request. If there is a source record, the information is included below. When a knowledge article is created from within another record, such as an incident, the source incident details are automatically populated within the knowledge article.

History Tab

The History tab shows auto-generated dates: Date Created, Date Updated, Date Published, and Date Retired. It also keeps track of the user who created the record and the user who last updated the record. Below these fields, the History Events section contains a row for each time a change was made to the record, either by a user or a rule, and which fields were modified. For instance, in the screenshot below, the user with login devadmin updated the field Special Compliance Requirements on June 30, 2016 at 15:31.

Knowledge Article Processing and Subscriptions

This section describes how to process and subscribe to knowledge articles.

Creation

A knowledge article can be created by clicking New in the action bar of the main table view, or by clicking Create Knowledgebase Article within a Closed incident or service request, Completed change request, or Resolved problem. If created within an incident, service request, change request, or problem, the source record information is automatically populated into the knowledge article. In addition, some of the information from the source record is mapped into knowledge article fields, such as the article Title, Purpose, Attached Files, and Working Notes. Knowledge Articles start in a default Status of Draft.

For example, if Create Knowledge Article is clicked within a Closed incident, a new knowledge article record appears with the following fields pre-populated:

Knowledge Article			

Save **Cancel** | Details | Progress | Services | Source | Emails | History | « »

| ID | 55 | Status | Draft ▾ | **Cancel this Document** |

*Title | Internet connection is slow

Assigned Team | Document Management Team ▾ | | Assigned Person | ▾

Submit for Review | | **Submit for Approval**

Document Details

*Knowledge Type | Choose one ▾ | | Knowledge Subtype | ▾

Topic(s) | 📋

Priority | Choose one ▾ | | Rush | ○ Yes ● No

Purpose | This problem happens throughout the day. How can I improve my performance?

Content | Please try running the following command: In your Start menu, type in cmd to bring up the dos command prompt. Type in traceroute and then send us the result.

Edit

*Special Compliance Requirements | ☐ HIPPA ☐ SOX ☐ ISO20000 ☐ Copyright ☐ Other ☐ None

Compliance Notes |

*Document Access | Choose one ▾ | | Audience | ☐ Customers ☐ Partners ☐ Prospects
☐ Employees ☐ Vendors ☐ Executives
☐ Media ☐ Analysts

Document Files

Source Files | Attach/Manage
1. sample-agreement.docx

These fields can be modified as needed. Additional required fields will also need values before saving the article.

Review and Approvals

Once an article has been created in a Status of Draft, it can be submitted for review from the Document Management Team by clicking Submit for Review. This changes the Status of the article to Pending Document Manager and sends an email to the Document Management Team. A document manager must then review the information, edit information if necessary, and either submit the article for approval or publish without approval. Alternatively, if the document manager determines that the Submitter must review the article or make additional changes, the manager can click Return to Submitter.

The field Requires Approval on the Progress tab indicates whether the article must be approved. The default value for this field is Yes, but can be set to No manually if approvals are not needed.

If approvals are not needed, the document manager will set Requires Approval to No and then can click Publish Without Approval. Clicking this button changes the Status of the article to Published.

If approvals are needed, the document manager will leave Requires Approval set to Yes and select the users who need to approve from the Approver(s) lookup:

The approver(s) selected will appear in the Approver(s) field, and then the document manager will click Submit for Approval. This changes the Status to Pending Approval and generates an approval record for each of the approver(s) selected, shown in the Approvals table below:

Agiloft System and Ralph Knowles will each get an email asking them to review and approve the document. By default, any approver can approve at any time relative to other approvers, but this can be modified by giving the approvals a sequence. The approvers can click the hyperlink in the email to view the approval record, where additional details about the article can be accessed by clicking on the Document Title.

Approvers can Approve, Require Changes for, or Permanently Reject the article. Any approval notes entered by approvers will appear in the Approval Notes field in the article. Once each approver has approved the article, the Status of the article is automatically changed to Ready for Publication. In addition, the Submitter and the Document Management Team receive an email alerting them that the article has been approved.

A document manager may then click Publish, which changes the Status to Published. When an article is published and the Document Access is Public or All Internal Users, internal customers will be able to see the article. In addition, if a user has subscribed to one of the Topic(s) for this article, he will receive an email about the new article.

Revising Knowledge Articles

If a document manager wants to create a revision for an existing knowledge article, there are two options: clone the article for the revision or create a service request for the revision.

To clone the article, click Clone Article for Revision. A new knowledge article record will appear with some information mapped from the source article, including Title, Assigned Team, Purpose, Content, Special Compliance Requirements, Compliance Notes, and Source Files. Additional information can be entered and pre-filled fields can be modified before saving the revision record. The default Status of the revision article is Draft.

To create a service request for a revision, navigate to the Revisions section on the Progress tab and click Raise

a Service Request for Revision.

This will bring up a new service request record with Source Files is mapped into the service request's Attached Files. In addition, the Summary of the service request is set to "Revision of [knowledge article title]", the Business Service is set to Printing and Document Services, and the Service is set to Change to an existing knowledge article.

An Impact and Urgency must chosen prior to saving the new service request. In addition, an explanation of the revision should be added to the Description field. More information about the knowledge article being revised is visible on the Related Records tab. The source knowledge article is populated automatically.

In the knowledge article, the service request for revision is automatically appended to a table of Service Requests on the Progress tab.

Retiring Knowledge Articles

After an article is Published, if it is no longer needed, it can be retired by a document manager using the Mark Retired button. Clicking this button changes the Status of the article to Retired.

Knowledge Topics

Each knowledge articles can be associated with one or more topics, such as Security, HR, or Confluence. This categorization makes it easy search for knowledge articles for certain topics. As mentioned above, users subscribed to a topic, such as Confluence, will receive an email when an article about Confluence is published.

Users can subscribe to topics by clicking on My Profile in the end user interface:

At the bottom of the first tab, a dropdown of topics is visible. All topics to which the user has not subscribed will be listed in the dropdown. To subscribe, the user will select the topic and click Subscribe to Topic.

The screen will then refresh and show the topic listed as one of the user's topics. In addition, a popup will appear to inform the user know that their subscription was successful.
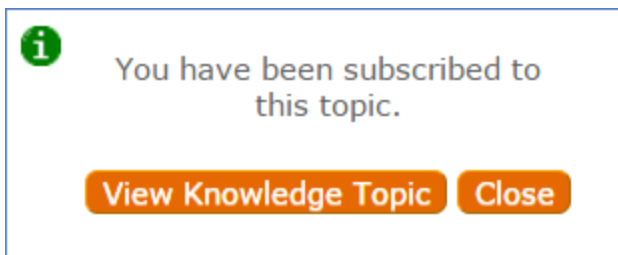
Rating Knowledge Articles

Any user who reviews a knowledge article can give it a rating by clicking Rate this Article. A new record will appear where the rating information can be entered.

Workflow

The following diagram represents the different knowledge article states and the allowed transitions between the states:
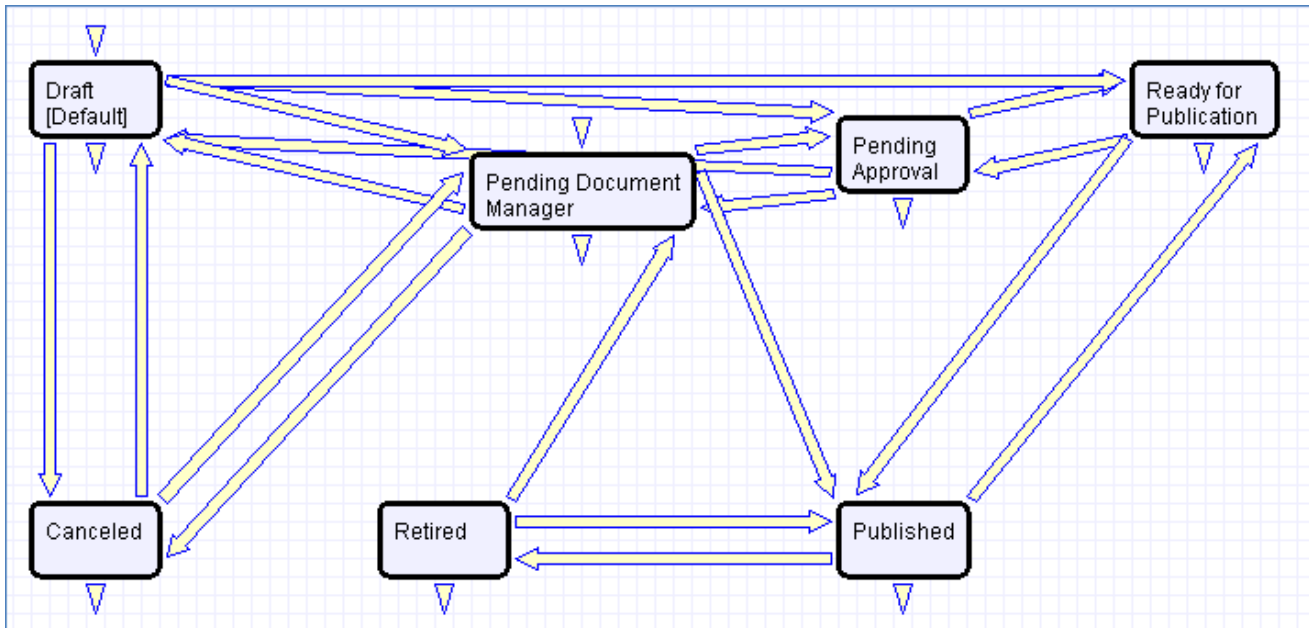
Reports

Knowledge Article reports are listed below:

Evaluating Knowledge Article Usage

Information about how many users have viewed a knowledge article in a certain time can be captured. Several reports have been created to summarize this information. For example, one chart shows the number of file views per knowledge article each month:

Jul 05 2016 14:29:23

Knowledge Article File Views in Past Month

Number of Activity Log: 4

Legend: confluence-user-guide.docx, sharepoint-user-guide.pdf, user-guide.docx

Additional details on who viewed the file and when are part of this report as well.
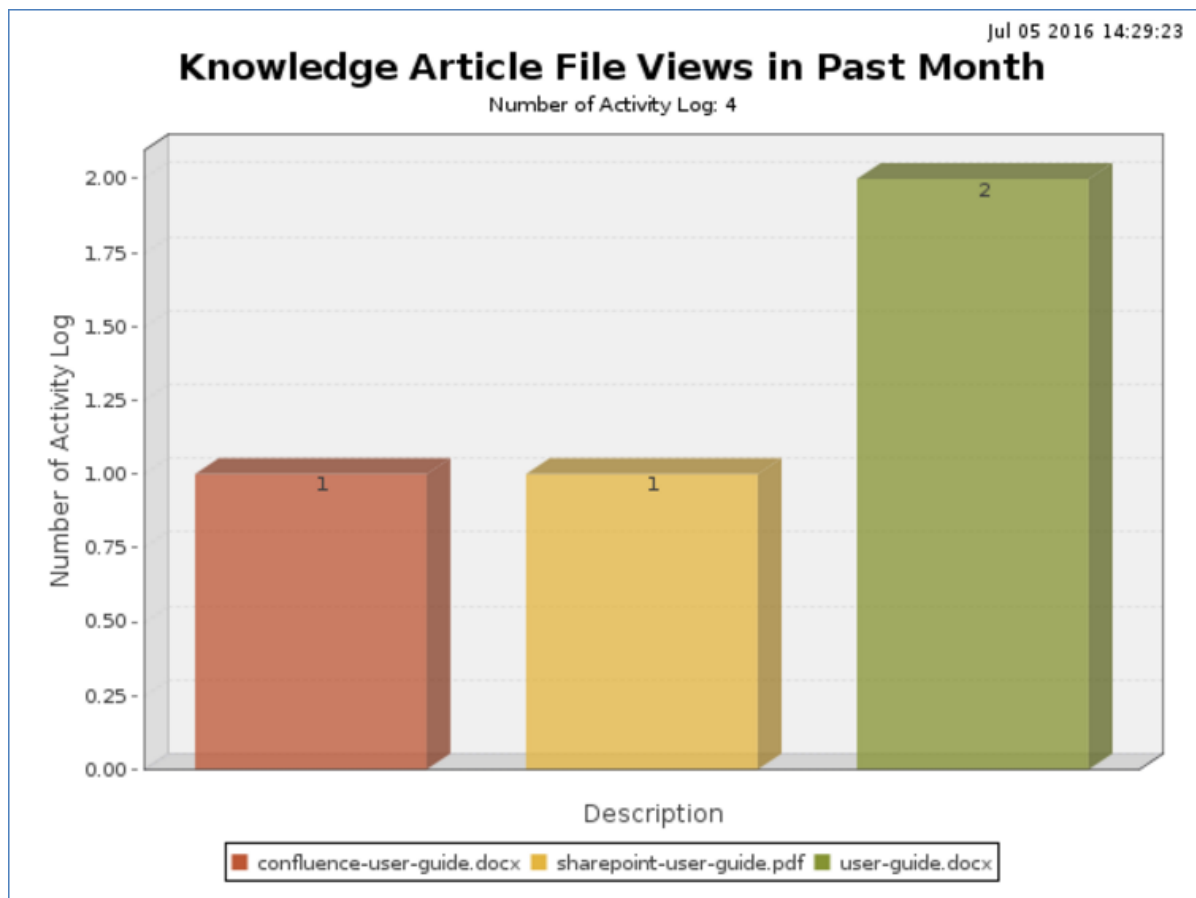
In addition to looking at file views, record views can also be included in reports. This information can be used to assess what types of knowledge are most useful or have been relevant over the past month.

Evaluating Knowledge Article Need

Reports in other tables can also provide information relevant to knowledge. For instance, this report of Closure Categories for incidents gives information on what types of incidents are occurring most frequently and what kinds of improvements can be made. Again, more details about the data used to generate this report are readily available in the HTML portion.

Checking for Similar Knowledge Articles

To help identify redundant information, as well as to provide information about what types of knowledge are being published, there is a report called All Active Knowledge Articles, grouped by Knowledge Subtype:

All Active Knowledge Articles, grouped by Knowledge Subtype

Number of Knowledge Articles: 27
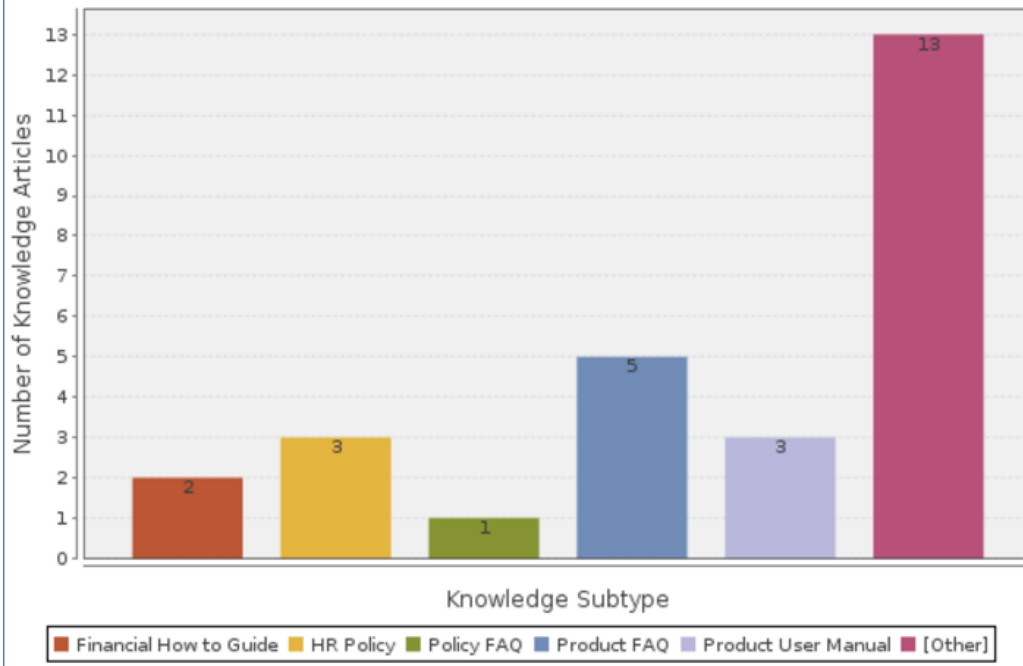
The more detailed list of articles published within the same Knowledge Subtype can be used to easily determine which articles may be able to be combined.

Background Tables

There are several other tables essential to the system that hold data that is less active and workflow-oriented than the request tables. These tables are briefly described in this section.

- Companies Table
- People Table, Employees and External Users Sub-Tables
- Calendars Table
- EUI Templates Table
- Replacement Variables Table

Companies Table

Purpose: This table holds information about companies that interact with your organization. It may include customers, vendors, prospects, manufacturer, and so on.

Use Case

Companies may be created by conversion from the Leads table. Members of the Admin, Professional Services, Support Staff and Sales groups may also create new Company records directly. For contract management users, companies can be created with an action button when filling out a new Contract record.

The Company table contains mostly static data, and thus does not have any associated workflow actions.

Information about a company and a company's relationship to your organization is stored in a Company record. This information can be everything from the company's billing address to information about their support contracts with your company. More complex data relationships are displayed via linked fields, which pull any records from the Contacts and Opportunities tables that reference the Company record. This allows you to see any business deals or transactions in progress and keep track of any Company employees that your organization contacts.

Company address information is stored in the Locations background table and displayed on the Locations and Contacts tab. A parent company may have several locations, e.g., a billing office, branch locations, and headquarters.

As a background table, many other tables link to the information stored in the Companies table. The Related Records tab shows related tables for Support Cases, Contracts, Insurance Certificates, and Assets.

New Insurance Certificates are typically added from the Company record. An Insurance Certificate Owner is defined just above the Insurance Certificates related table. The Insurance Certificate Owner is notified fourteen days before one or more certificates is due to expire, provided that there is at least one active or pending contract. In the certificate record, the Main Contact is the primary contact at the vendor company.

Insurance Certificates

| Insurance Certificate Owner | Maurice Sendak | 🔍 | | Insurance Cert Owner Email | example26@example.com | 🔍 |

Insurance Certificates

Status: 1 record(s) found, 1 pages. Click here to re-count records.

New | Mass Edit | Delete | 🔍 Unlink | Actions ▾ | Views ▾ | Search 🔍 ▾ | ✉ ▾ |

	Edit	ID ↓	Type of Certificate	For Company Name	Status	Amount of Coverage	Expiration Date	Attached Files
☐	✎	14	Excess Liability	Acme Micro	Valid	$35,500.00	May 27 2016	

| No. of Active Insurance Certificates | 1 |

Insurance certificates added to a company record are automatically linked to all contracts associated with the vendor company. See the Insurance Certificates Table for more information.

Ownership

Records in this table are "owned" by the individual assigned sales rep, so each record is associated with a particular user login. Admins and members of the Contract Owner, Contract Manager, Marketing, Project Manager, and Sales groups can view and edit Companies. Most groups can view their own Company, and most internal users can view others' Companies.

People Table, Employees and External Users Sub-Tables

The User table tree is used to store information about individuals who use your system.

The People table contains data about individual users of your system and individuals with whom your organization has contact or business relationships. There are two basic subtypes of Users: External Users and Employees. Most external Contacts will go in the External User table, as the Employee table tracks information that does not apply to people outside your company, such as working hours and hire date. People records normally have a link to a record in the Company table to define which Company is related to the Person. By default, People records have an embedded list of records from another table, called a related table, listing all of their Support or Helpdesk Case records (tickets).

People are considered static in that they do not change status. By default, no workflow is specified for the People table. People may be created manually by Guests, Support Staff, and members of the Professional Services, Sales staff and Admin groups. Guest record creation is enabled by default so users can create their own logins using the limited-access "register" account. Records created by the Guest login "register" are added to the "Selfregistered users" group automatically by a rule.

People may also be created as the result of a conversion from a Lead record, or may be created as part of an import from an external database such as an LDAP directory. Each Person record is assigned either a Team or an individual from the Employees subtable as the "Owner".

The Employee sub-table has a rule attached to it that runs a conversion of an Employee record to the Approvals table when triggered by a field changed on the Employee record. This facilitates the Documents approval process.

Calendars Table

This table contains event records displayed in the Calendar block in the left pane, or using a normal table view.

Special Use Case

The Calendar table is a special table used in the Staff interface Calendar pane feature. Records in this table display as events in the Calendars for the individuals and groups that they reference. Calendar records are mostly static and do not have any default associated workflow, rules, email setup, saved searches or charts. The Calendar pane is only available to Staff users, and cannot be made available to End Users.

By default, only members of the Admin, Professional Services and Support Staff groups may create Calendar records, and Professional Services and Support Staff may only view their own. No other groups have access to view Calendar entries by default, so other staff groups using the Calendar pane must be individually activated or given view and edit permissions. Records in this table are "owned" by their creators, and are linked to the creator's login.

EUI Templates Table

This table contains HTML files used in creating a customized End User portal. It has sample files that can be used, and you may create your own html files and upload the html content to records in this table to serve up the pages.
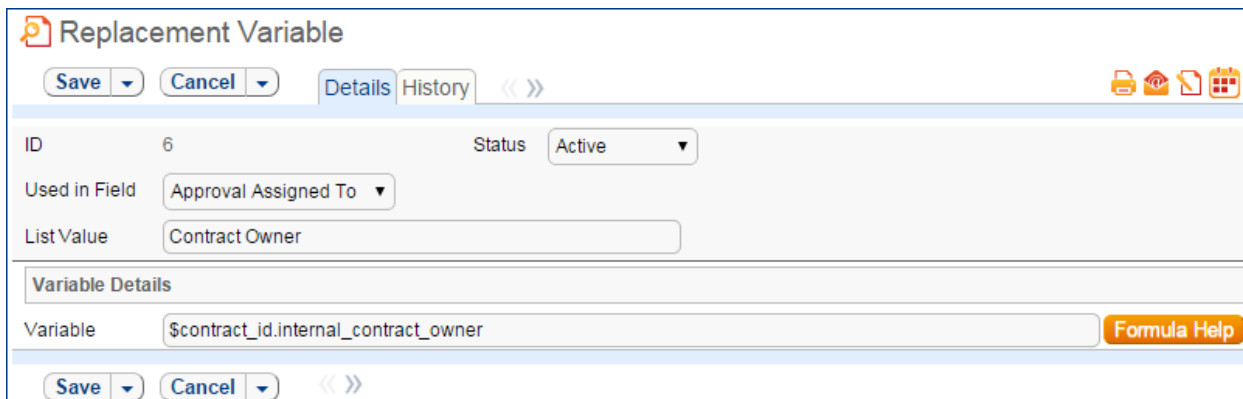
The EUI Template table is a special table used for storing HTML files used in changing the presentation of the Agiloft End User Interface. Since these files undergo little or no change during their lifetimes, no workflows, rules, or actions are associated with them, and no charts, reports or saved searches exist for this table.

EUI and Documents Table

When the Documents Table is enabled, the EUI can be configured to allow access by end users to the table. In order to enable this, the "main.php" and "menu.htm" files need to be edited so that all references to "global.menu.file" and "global.home.file" are "globoal.menu-doc.file" and "global.home-doc.file", respectively. These references can be found in '#ew_include("$ewText.get('')")' calls in those html pages.

Replacement Variables Table

The Replacement Variables table is a background table that can be used to support any of the process tables included in the Standard System KB, such as Approvals, Contracts, Projects, or Tasks. Replacement Variable records store a variable chain used to pull in values from a linked field relationship. Generally, these variables are copied into template records (e.g., in the Approval Templates table) and then used to dynamically assign records or set other field values when a process table record is generated from a template.

Replacement Variables are used by Approval Templates to generate Approval records for contracts, change requests, documents, or service requests.

To create a Replacement Variable that will be used in Approvals, set Used In Field to the value "Approval Assigned To". The value in the List Value field will shows up in the Assign To drop-down in an Approval Template. Last, create a **variable chain** from the Approvals table to the appropriate field variable. For example, $contact_id.requester_name finds the name of the Requester in the linked Contract record, while $contract_id.requester_id.manager_name finds the Requester's manager.

> The Variable should be a variable chain that starts from the Approval record—not from the Approval Template!

Note: The values "Task Assigned To" and "Other" do not link to any automation in the default setup, but they can be deployed and used with minor modifications to Task Templates or other template table. To use a Replacement Variable in an Approval Template, staff users can create an Approval Template and set Assign Approval Based On to the value "Person from Contract".

The user can then select a value for the Assign To dropdown, which contains a link to all Replacement Variable records for which Used in Field is "Approval Assigned To". When an Approval is generated from that Approval Template - see the Workflows Table and Handling Approvals sections - the Approver is set using the variable chain in the Variable field from the Replacement Variable record.

Additional Functions

In addition to the tables and functions described above, several additional functional tables are included with the ITIL KB that may be activated for use depending on the functions you requested when signing up for a free KB or downloading and installing. They cover the general functions of Contract Management, Project Management, and CRM (customer support, sales, and marketing).

These include the following main tables in addition to their supporting tables: Contracts, Documents, Projects, PO's, Surveys and Survey Answers, Leads, Opportunities, Quotes, Products, and Campaigns.

- Contract Management Tables
- Documents Table
- Project Management Tables
- Support Cases Table
- SalesCRM Tables - Leads Table
- Sales CRM Tables Opportunities/Deals Tables

Contract Management Tables

The Contract Management system is designed to be ready out of the box with a variety of approval processes built-in. It is easy to eliminate or modify any functions to better fit a desired business process. Some time-based rules are disabled by default and need to be enabled for the table to work correctly. To do this, go to the Rules tab of the Setup Contracts page, edit a rule with "(Disabled)" in the title, and click "Yes" in the "Rule is enabled" section at the bottom of the page.

This section describes how the default contract management system is configured. The Contract Management system is comprised of a few main tables with background tables playing a supporting role. The main tables are described in the following pages.

- Contracts Table
- Insurance Certificates Table

Contracts Table

The Contracts table holds all contract records. It also controls all associated automation and notifications related to contracts. A representative record is shown below.

Required fields, marked with a red asterisk, are: Record Type, Internal Contract Owner, Contract Type, Days in Advance to Notify for Renewal, Contract Title, Contract Description, Contract Start Date, and Contract End Date.

Staff Use Case

This section covers the use case for staff members inside Agiloft.
Each record in the table holds information about an individual contract including details about the contracting party, approval information, attached contract file and supporting documents, and renewal details.

Contract Creation

Contract records may be created by members of the Admin, Contract Creator, Contract Manager, and Contract Owner groups.
Contracts may be created in one of two ways:

- Click New in the Contract table action bar
- Use the Create Related Contract button on the Renewal / Related Contracts tab to create a renewal, subcontract, or amendment from the current contract. This button is available only if the contract is in a Status of Signed, Active, Expired, or Canceled. Creating a new contract with the Create Related Contract button will automatically link it to the current contract by populating the Parent Contract ID field in the newly created record. For a more detailed explanation of creation by this method, see the Handling Related Contracts and Renewals section.

Contracts fall into one of four categories:

- Contracts
- Master Agreements
- Subcontracts
- Amendments

The Record Type field in the common area indicates the contract's category. "Contract" is the default Record Type for newly created contracts. It can be used to indicate either a stand-alone contract or a contract that exists under a master agreement. In the latter case, the Parent Contract ID can be filled in upon creation of the contract. Subcontracts and Amendments will be linked to a parent contract automatically, if they are created by clicking the Create Related Contract button in the parent contract record.

 Below the Record Type are fields storing the Assigned Team and the Internal Contract Owner. The default Assigned Team is the Contract Manager Team. The Internal Contract Owner is the person responsible for overseeing the contract and ensuring timely renewals and approvals. The list of available owners is filtered to users who are in the Contract Owner Team or the Contract Manager Team. The default Internal Contract Owner is the user who creates the contract record, provided that user is in the Contract Manager or Contract Owner Team. Users in the Contract Manager or Admin group can manually change the Assigned Team and Internal Contract Owner if needed.

Once the appropriate fields are filled in, the contract may be saved in a Status of Draft. The contract's Status field is changed automatically by the system at appropriate points in the workflow, generally when an action button is pressed or when some condition is met; however, users in the Admin group can manually override the Status if necessary.

Information about the contract requester, external company involved, and locations related to the contract may be added by clicking the lookup icon next to those fields. If a desired Requester, Company, or Location does not exist, a record must first be created in order to link it to the Contract.

Creating companies and contacts during contract creation

In the Contract Party Information section, users can find and link to an existing company, or create new ones as needed. To find an existing company record, use the lookup icon to search for the company. If the contract party isn't found, staff users can add a new company by selecting New Company. Enter the name and address information, then click Create Company.

Both company and location records are created in the background, and linked to the new contract when the record refreshes.

Similarly, you can create new entries in the people table if the main contact for the contract party is not an existing contact. Select New Contact, enter the information, and click Create Main Contact. Multiple contacts can be created in this way, and then the primary contact can be selected from the Main Contact drop-down.

When Contract Requesters submit a new contract through the end user interface, they can enter new company and contact information, but they don't see the buttons to create new records. When a contract is submitted for review, Contract Managers can edit the contract party information, or click the button to confirm the new company or contact and finish adding records.

This setup is intended to prevent end users from creating duplicate companies with slightly different spellings and to ensure that contract managers have ownership and control of this data. Permissions can be changed easily to allow end users to use the buttons to create companies and contacts directly.

Note that it is important for the contract managers to evaluate the new additions and either replace with existing data or click the buttons to create the appropriate background data. Otherwise, the linked data can't be used in print templates and emails.

Creating contract attachments from default print templates

The Attachments tab shows all attached files related to the contract and provides the user with options to create and edit related attachments. For certain contract types, when the Document Source is set to Standard Template or Modified Template, the Print Template to Generate field appears. The Document Source can also be set to 3rd Party or Internal – Other to indicate a document provided by the external contract party, or which has been internally generated but not from a template, respectively. The Print Template to Generate has a default value set based on the contract type. The Create and Attach action button is used to auto-generate a contract document from the Microsoft Word print template specified in the Print Template to Generate field and stored in the Print Template File field. Clicking Create and Attach generates the document and creates an attachment record to hold the attached file. It also increments the Version Number field.

If the Contract Party Information section is not properly filled out and linked to records, a validation action warns the user to finish creating new records before generating the attachment.

Creating contracts and attachments from inbound emails

Attachments can be automatically generated from inbound emails containing attached files. Files from inbound emails are mapped to the Transitional Contract Files field in the contract, assigned a predetermined Attachment Title and Attachment Type value, and then converted into a new attachment record. The file is mapped into the Attached File field in the attachment record. If multiple files are attached to an inbound email, each one is converted into a separate attachment record for that contract.

Working with attachments

The All Contract Attachments related table is also used to manually create and attach files to the contract, such as performance bonds, signed agreements, or other documents from the contract parties. Create attachments from within the contract record by clicking New on the related table's action bar.

The Select Files > Add/Remove from DS Envelope actions on the action bar can be used to add or remove selected attachments to and from the DocuSign envelope when using DocuSign integration, and if integrating with Adobe Sign there are analogous actions. When these actions are selected, the To Be eSigned field is set to Yes or No, and the DS Files to Sign field on the DocuSign tab is updated with the attachments. The Select Files > Supersede File action sets the Status of the selected file to Superseded.

For more information on Attachments, refer to the Attachments Table section.

Tracking insurance certificates

The Attachments tab also holds a related table that can be used to track insurance certificates separately from other types of attachments. Certificates can be found directly in the Insurance Certificates table, and in related

tables in both the Company and Contracts tables.

When you add an insurance certificate to a company record, it is automatically linked to all pending or active contracts with that vendor by a daily time-based rule (disabled by default).

To create a certificate from a contract record, click New on the related table **action bar**; the certificate will be automatically linked to the contract. You will need to look up the company vendor in the For Company Name field when creating certificates from a contract.

The Insurance Certificate Owner (defined in the company record) is notified fourteen days before an insurance certificate's Expiration Date. When a certificate expires without being renewed, the Contract Managers of any related active contracts are notified on the date of the expiration.

For more information about insurance certificates and related automation, see the Insurance Certificates Table section.

Another method for managing insurance certificates

You may prefer to manage insurance certificates as contract attachments, instead of using the Insurance Certificates table. For example, if Contract Managers are responsible for renewing and maintaining insurance certificates, it may be best to use the Attachments table to hold the certificates, along with all other types of attached documents for contracts. Managing insurance certificates as contract attachments means the Contract Manager will receive email notifications instead of the Internal Certificate Owner.

If this method makes the most sense for your particular business, you can remove the default Insurance Certificates field from the contract layout. Then, go to the Attachment Type table and edit the record called Insurance Certificate. By default, this record's Status is Inactive; change the Status to Active to make it visible in the Attachments table.

Adding notes and sending emails

Notes relevant to the general contract process are entered into the General Notes field on the Details tab, and notes pertaining to the approval process are entered into the Approval Notes field on the Approvals tab.

Approvals								

Workflow Title   Dept. Head and Legal Sequential Approval

Launch Approval Process

Recheck Conditional Approvals

Approvals Needed

Status: 2 record(s) found, 1 pages. Click here to re-count records.

Actions ▼   Views 🐞 ▼ | Search 🔍 ▼ | 🗋

	Edit Link	ID ↑2	Step Number ↑1	Approval Title	Approval Team	Approver	Status	Approval Notes	Approved/Rejected By	Date Approved/Rejected
☐	📝	607	1	Department Head Approval	Contract Manager Team	System	Queued	Looks okay now.	System	Jul 23 2015 12:28
☐	📝	606	2	Legal Approval	Legal Team		Requires Change	I still think the contract needs changes to some of the terms.		

Number of Approvals Needed   2   Number of Approvals Completed   0

All Notes by Approvers

Approval Notes

> [System Jul 23 2015 12:30]
> I still think the contract needs changes to some of the terms.

> [System Jul 23 2015 12:28]
> Looks okay now.

> [System Jul 23 2015 12:26]
> Please review this contract again and make changes.

Emails can be sent to Internal Contacts and External Party Contacts from the Emails tab. First choose the type of Recipient(s) (Internal Contacts, External Party Contacts, or External Party and Internal Contacts), then click the lookup icon to select which contacts to email. Fill out the Email Subject and Email Text fields as needed, select any files to send with the email, then click Send Email.

Renewals and previous contracts

Information about the renewal process is stored on the Renewal / Related Contracts tab. Fields for capturing the Renewal Notification Date and Days in Advance to Notify for Renewal are provided as a default. If relevant, information about the renewal contract or previous contract are automatically updated by the system. Users typically do not manually link contracts.

Related assets

Assets can be linked to individual contracts from the Assets tab. Assets must be added separately to the Assets table before they are available to attach to contracts.

Depending on the selection in the Asset Involvement field, different fields appear for the user to input linked assets.

For hardware assets, select "For One or More Assets". For software assets, choose "For Software Application". For contracts with no linked assets, use "No Assets" from the Asset Involvement drop-down.

Signing contracts with DocuSign

A Signature tab appears to the right of the Assets tab. This tab contains fields for contract document Signers, and contains all fields related to DocuSign. The DS Files to Sign field holds attached files from the Attachments table that have a To Be eSigned value of Yes. The Refresh Files action button refreshes this field in case changes to the attachments were made in the same session under the contract's Attachments tab.

Fields under the Signers heading are used to populate the signature page of the contract's print templates and to add DocuSign tags if the contract uses DocuSign.

Once the contract is ready to sign, use the Create DocuSign Envelope action button to create a DocuSign Envelope record and attach the files held in the DS Files to Sign field. It also creates a DocuSign Recipient

record for each signer. These records are then shown in the related tables for DocuSign Envelopes and DocuSign Recipients on the Signature tab. Only users in the Admin and DocuSign Users groups can access the DocuSign fields on the Signature tab. For more information on DocuSign, refer to the detailed DocuSign User topics.

Handling Approvals

The order and nature of approvals depends on the Workflow Title selected on the Approvals tab. Selecting a Contract Type in the common area filters the available Workflow Title choices. For information on setting up the individual Workflows, refer to the Workflows Table section.

To submit a contract for approval, first select the correct Workflow Title from the drop-down provided. Click Create Approvals to generate a set of Approval records in a Status of Queued. Once the records have been created, click Launch Approval Process to update the Status of the contract to Pending Approval and update the Status of the approval(s) with the lowest step number to Pending Approval. There may be more than one approval in the lowest step number for parallel approvals.

The system automatically notifies the first approver in the sequence. A progress bar also appears in the common area to provide a quick visual reference of the approval process.

To advance the contract workflow, an approver will use one of three action buttons to change the Status of the approval record:

- Approve to send the contract to the next approver in the sequence.

- Require Changes to send the contract to the previous approver for changes.
- Permanently Reject if the contract requires significant changes.

Both Require Changes and Permanently Reject require the user to enter notes in the Approval Notes field. These notes are appended to the Approval Notes field in the contract record and are also viewable from any other approval record linked to that contract. As each approval is approved, the system notifies the next approver in the sequence that a contract is pending their approval. A list of all approval records is displayed under the Approvals Needed subsection and automatically updated as approval records are modified.

If a user who is not on the current approval team attempts to approve a contract, the system recognizes the error and prevents the user from completing the approval action. When all the required approvals are received by the system, the contract Status is automatically changed to Approved.

Handling Related Contracts and Renewals

Related contracts are handled in the Renewal / Related Contracts tab of a contract record. To create a related contract, select the New Record Type of the contract. The choices in the drop down are filtered by the Record Type of the originating contract. These fields are visible only when the contract's Status is Active, Canceled, Expired, or Signed. Clicking Create Related Contract maps relevant information from the current contract into the new one.

If the new contract is a renewal, click Create Renewal Contract on the Renewal / Related Contracts tab of the parent contract. This button is only visible when the contract's Status is Active, Canceled, Expired, or Signed. A new contract generated in this way can be edited before saving.

If applicable, the system automatically links renewal contracts to any preceding contracts, creating a chain for auditability. Any assets linked to the preceding contract will be linked to the renewal contract as well. This information and related information on the contract process for renewals are stored in the Renewal / Related Contracts tab. Users typically do not enter information into the Renewal Contract and Previous Contract Information fields manually.

Contract Processing

This section covers the remaining Status changes not mentioned in the explanations above. Once a contract changes to a Status of Signed, the system automatically updates the contract to a Status of Active when the Contract Start Date occurs. Similarly, the Status is changed when the Contract End Date occurs. If the contract does not have an associated renewal contract, the Status is automatically set to Expired; if the contract does have a renewal, the Status is set to Renewed. If the contract is auto-renewing, the Contract End Date is increased by the Renewal Term in Months and the contract remains Active.

To cancel a contract, click the Cancel Contract button in the common area. Additionally, users of the Admin group can manually change the contract status to Canceled.

Turning Off Approvals

Contract Management tables have Approval handling set up by default. The associated processes may be turned off in order to use Agiloft as a contract repository. To turn off Approvals, do the following:

1. Edit field permissions to allow the Contract Manager Group to edit the Contract Status field in their own contracts and in others' contracts.

2. Remove status-changing buttons from the layout: Submit for Approval and Mark as Signed.
3. Remove the Approval tab and related fields from the layout.

For help on configuring table layouts, please refer to the Administrator Reference Manual or online help.

End User Use Case

This section covers the use case for end users in a Contract Management context. Members of the Contract Creator group are internal employees accessing the system via the end user interface. Contract creators, also called contract requesters, are users who submit requests for contracts but do not work on other users' records. Below is a representative home page for an end user in the Contract Creator group.
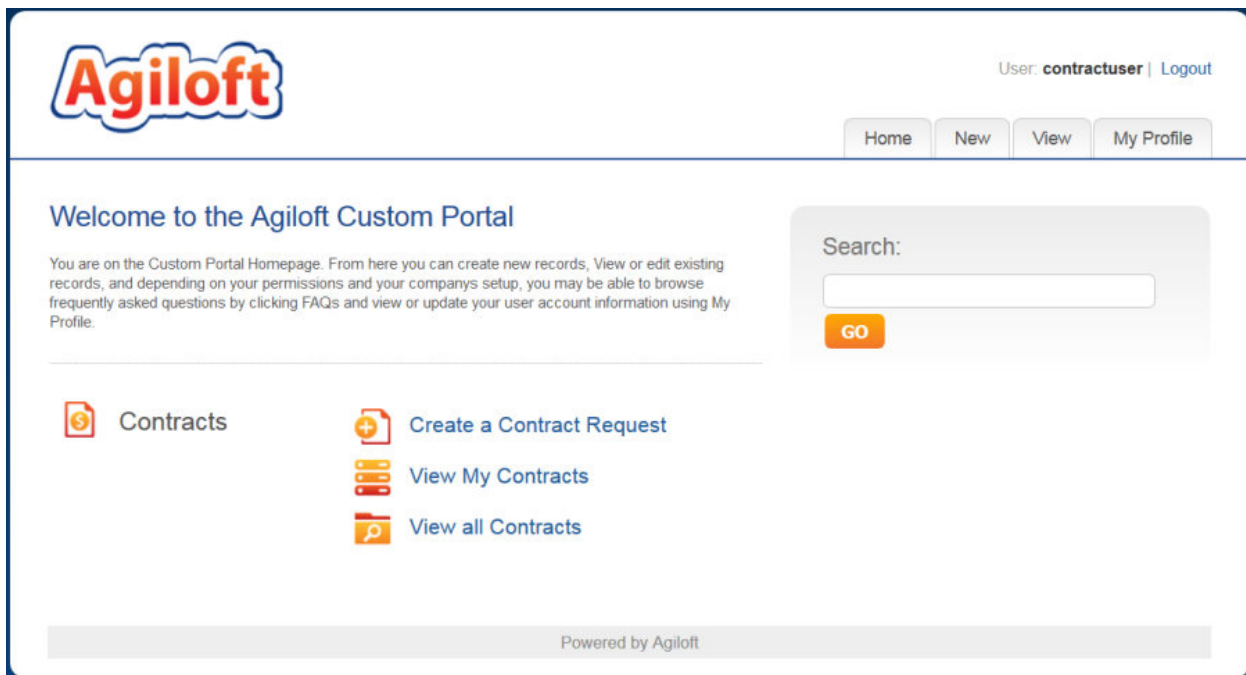
Contract Creation

Users in the Contract Creator group may create contracts by clicking the Submit a Contract link or tab on the home page. A simplified contract form is presented to the end user. Many of the fields are hidden from the layout or restricted by field-level permissions.

In the Contract Party Information section, users can find and link to an existing company and contact, or create new ones as needed. Select New Company and then enter the name and address information. After the contract is submitted for review, Contract Managers can confirm the new company to finish adding a record. Similarly, end users can suggest a new company contact by filling out the fields in the Party Main Contact section of the form.

Once the required information is filled in, the contract can be saved for later revisions. The contract requester can also press the Submit for Review button to request approval from a Contract Manager. Contract requesters can be contacted to update the submitted contract, but they are typically no longer involved in the

approval process from this point forward.

After the contract requester submits the contract for review, a contract manager decides whether to continue with the approval process or reject the contract request.

Working with Contracts

At any time, the contract requester can view contracts they previously submitted by clicking the My Contracts tab. Contract Creator group members can edit select fields in contracts they own. Certain fields such as Contract Amount, Contract Start Date, and Contract End Date (among others) are not editable by the contract requester if the contract's Status is Pending Approval, Approved, Signed, Active, Renewed, or Expired. This is to prevent changes to currently active or in process contracts.

The contract requester can view all contracts they have permission to see by clicking on the All Contracts home page link or tab.

Ownership

Records in this table are owned by the Contract Requester. Specifically, a record is owned by the user whose ID matches the number in the Requester ID field. By default, the Contract Requester is the user who created the contract record.

Workflow

The Contracts table has the following default workflow:
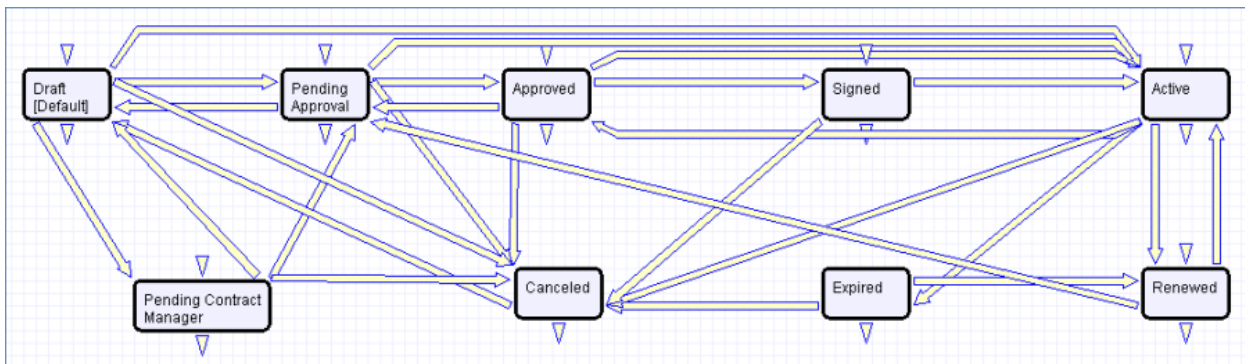
Reports

The Contracts table contains the following default Charts/Reports:

Charts/Reports

Status: 8 record(s) found, 1 pages. Click here to re-count records.

New | Copy | Delete

	Edit	View	Title ↑	Filter	Published
☐			All Active Contracts with Vendors	Active Vendor Contracts (C)	No
☐			All Active Customer Contracts	Active Customer Contracts (C)	No
☐			Contract Amount by Status		No
☐			Contract Expirations in next 90 days	Active Contract expiring within next 90 days (C)	No
☐			Contracts by Party Type		No
☐			Contracts by Type		No
☐			Contracts in Progress	Contracts with an In Progress Status (C)	No
☐			Value of Signed Contracts starting in Past year	Active Contracts whose start date is less than a year ago (C)	No

Insurance Certificates Table

This table holds insurance certificates linked as attachments to Companies and Contracts. Each record in the table represents one insurance certificate linked to a unique vendor company, but may be linked to one or more contracts.

Use Case

Insurance certificates can be created in one of three ways:

- From a Company record, using the related table on the Related Records tab (preferred)
- From a particular Contract record, using the related table on the Attachments tab
- From the Insurance Certificate table **action bar**

New Insurance Certificates are created with a default Status of Valid. They may also have a Status of Expired or Contract Inactive. The Expiration Date and Main Contact are required fields. The Main Contact is the person at the vendor company from whom a replacement certificate can be requested. You can customize the Type of Certificate choice list as needed; the default options include Auto, Worker's Comp, Excess Liability, General Liability, etc.

Insurance certificates are primarily linked to vendor companies in the Company table. Once you add an insurance certificate to a vendor record, it can be linked to any contracts with that vendor. Insurance certificate records contain information about related contracts for reference, held in the Contracts field. A time based rule, disabled by default, is designed to update the Contracts linked field with all contracts for that vendor that are currently pending or active. As a result, insurance certificates are automatically displayed within those contracts.

An Insurance Certificate Owner is defined in the company record just above the Insurance Certificates related table. The Insurance Certificate Owner is notified fourteen days before one or more certificates is due to expire. When the Expiration Date arrives, the certificate's Status is updated in one of two ways:

- If any of the associated contracts have a Status of Active, the system updates the certificate's Status to Expired and an email notification is sent to all the Contract Owners about the expired insurance certificate.
- If none of the related contracts are Active, the system updates the certificate's Status to Contract Inactive.

Ownership

Insurance Certificate records are owned by their creator. Specifically, an Insurance **Certificate** record is owned by the user whose Login matches the Creator Login field.

Automation

The Insurance Certificates table has the following rules set up. A rule running on the Company table sends notifications about upcoming expiring certificates.

TB: Update Contracts (daily) (disabled)

Purpose: This rule updates the linked field of Contracts to find all contracts related to the insurance certificate.

Runs: Daily at 1:00 AM

Search Filter: R: Status is Valid

Priority: 0

Action:

- U: Set Default Contracts
    - **Updated Field:** Contracts
    - **Updated Value:** Use the Insurance Certificate to Contract: Status is greater than or equal to Active and Company matches search, and choose all records.

TB: Handle Insurance Certificate Expiration (disabled)

Purpose: This rule checks for **expired insurance certificates and then updates the certificate's Status. If the** related contract is still valid, the Contract Owner is notified of the expired certificate.

Runs: Every two days at 1:30 AM

Search Filter: Valid Certificates expiring TODAY

Priority: 1

Action:

- I: Handles Certs Expiring TODAY

```
if (Contract Status='Active') {
        Set Status to Expired
        Email Contract Manager of expired certificate
} else {
        Set Status to Inactive
}
```

Ownership

Records in this table are owned by the creator of the certificate. Specifically, a record is owned by the user whose Login matches the login in the Creator Login field.

Documents Table

Purpose: This table can be used to manage the creation and publication of documents of various types, from marketing collateral to employee procedure manuals. A light-weight parallel publication approval process is included.

Examples of documents that may be covered: FAQs, official memos, published company policies, user manuals, newsletters, press releases, and so on. The table may be used to manage documents that are accessed only through Agiloft (though the records in this table) or documents that are published at the company website, intranet or printed and distributed. Access to the documents is controlled through permissions based on a choice field within the record.

End User Record Submission

An end user belonging to the Document Creator team can create a document through the EUI. Action buttons will be provided to the end user to move the document through the workflow.

When a user submits a document for review, the contact information fields are automatically populated based on the details in his/her record - including the user's department. A direct link to the Department and the Department Manager will be auto-populated based on the submitter's department.

The record will be created in a default status of Draft. After supplying the required information and uploading a document, the user will click the Submit for Review button to begin the review process. The status of the record will be updated to Pending Review. If the user is not prepared to submit the record immediately, then can save the record and make further updates.

Technician Record Submission

Staff users in the Document Management, Admin and Document Creator groups can submit documents. Support Staff members may also convert to create a new FAQ Document from an Incident or Service Request. Only Admins and Document Managers can update the status of a Document record manually. All other groups will use Action Buttons to move the document through the workflow.

The record will be created in a default status of Draft. After supplying the required information and uploading a document, the user will click the Submit for Review button to begin the review process. If the user is not prepared to submit the record immediately, they can save the record and make further updates.

Processing of Records

When a document record is submitted, the Document Management Team is assigned by default and notified. A Document Manager reviews the document for content and formatting, and to determine if the document requires additional review.

If there are any issues with the initial document, the Document Manager clicks Return to Submitter to send the document back to Draft status. A rule then notifies the Submitter that the document requires revision. The user makes appropriate updates to the document and record, then clicks Submit for Approval again.

Handling Approvals

The Requires Approval field on the Progress tab determines whether approvals are needed before publication.

If the document does not require approval by document Reviewer(s), the Document Manager clicks the Publish without Approval action button. The Submitter is notified that the document has been published. If the Requires Approval field is set to Yes, a validation rule notifies the Document Manager that approval is required if the Document Manager attempts to publish without approvals.

Selecting Approvers and Creating Approvals

If the document requires approvals by Reviewer(s), the Document Manager will select the appropriate reviewers by adding reviewer names under the Potential Reviewers heading on the Progress tab. This field is a link to a single field (Full Name) with multiple values enabled in the Employee table, displayed as a multiple value box with a popup selection list and filtered to people on the Document Reviewers Team.

After selecting the appropriate Reviewer(s), the Document Manager clicks Submit for Approval, or manually changes the Status of the record to Pending Approval. A validation rule checks to see if the Document Manager has actually added reviewers to the form.

When the record is saved, an approval record is created for each reviewer by a conversion process from the reviewer's Employee record. A *linked record action* updates the Last Document ID field in the Employee record.

When the system detects a change in the Last Document ID field it performs a *data conversion action,* using the employee record to create an Approval record. The approval record is linked to the document record through the mapping of Last Document ID field to the linked Document set in the approval record. Additional reviewers can be added during the approval process; automation rules and actions will prevent the creation of multiple approval records for existing approvers.

The default Approval Status for new approval records is Pending Approval. When an approval record is

created, the reviewer is automatically notified that they have a document to review and approve. In the Approval record, the linked field set from the document record includes a hyperlink to the document using the view only source field display option for the Document(s) field under the Attachment heading. The reviewer can click the link to open the document, allowing the reviewer to mark up the document which can then be uploaded to the approval record (reviewers are not able to upload the document to the source record). All updates to the document will be done by the Submitter, the Document Manager or an admin.

Rejecting Documents

The reviewer may either approve or reject the document and provide Approval Notes. If the approval record is rejected, the reviewer must provide Approval Notes explaining why the document is rejected. A validation rule reminds the reviewer that they must provide rejection notes if a reviewer attempts to reject the approval without providing notes.

When any of the approvals are rejected, a rule in the Document table detects this via a calculation on multiple linked records field that keeps track of the number of rejections. When this field is updated by rejections, the rule sets the document's Status to Draft, emails the Submitter that the document requires an update, and sets a trigger field named Document Rejected, Requires Update in each of the linked approvals to Yes.

Setting the Document Rejected field to Yes in the approvals then triggers the rule below that sets the Approval Status field in all of the linked approvals to Requires Reapproval, unless that Approval already has a value of Rejected.

Since any rejected Approvals set the Status of the document back to Draft, the process essentially starts over. The Submitter can review the comments made by the reviewers and see any red-lined documents that have been attached to the approval records. Then the Submitter edits the document and presses the Submit for Review button again to change the Status of the Document to Pending Review and notify the Document Manager. The Document Manager can then keep the same Reviewer(s) or change them and Submit for Approval again. If a document is rejected by an approver and then accepted again (and the Status therefore changed from Pending Approval to Draft and back to Pending Approval), the status of all of the linked approvals is restored to Pending Approval, and all Reviewer(s) are notified by email.

Publishing Documents

Once all approvals are completed and the value of the field Total Number Still Awaiting Approval is 0, a rule sets the Status of the document to Ready for Publication. Both the Submitter and Document Manager are emailed that the document is ready for publication.

The Document Manager can then attach the final approved document to the Published Files field and update the Status to Published. A validation rule checks again to see if there are still any pending approvals.

Once the Document is published, an email notification is sent to the Submitter.

Document

[Save ▾] [Cancel ▾] | **Details** | Progress | Emails | History | ‹‹ ››

| ID | 10 | | Status | Pending Approval ▾ |

*Title | How do new employees login to our network?

Assigned Team [▾] | Assigned Person | Chris Connoly ▾

Document Details

Document Type	FAQ ▾	Document Subtype	Choose one ▾
For Department	HR ▾	Department Manager	Paula Ling
Priority	High ▾	Rush	No ▾

Purpose | For new employees to help them work until they have been set up.

Description | This FAQ explains the process to login as a guest user until your employee login has been created.

[Edit] ○ HTML ● plain text

Document Access [All Internal Users ▾]

Audience
☐ Customers ☐ Partners ☐ Prospects
☑ Employees ☐ Vendors ☐ Executives
☐ Media ☐ Analysts

Document Files

Source Files [Attach/Manage]
No Files Attached

Published Files [Attach/Manage]
No Files Attached

Published Format [PDF ▾]

Internal File URL [ ]

Related Documents

Related Document(s) 🔍

Submitter Information

Submitter Mia Lazano 🔍 | Submitter Department HR 🔍

Workflow
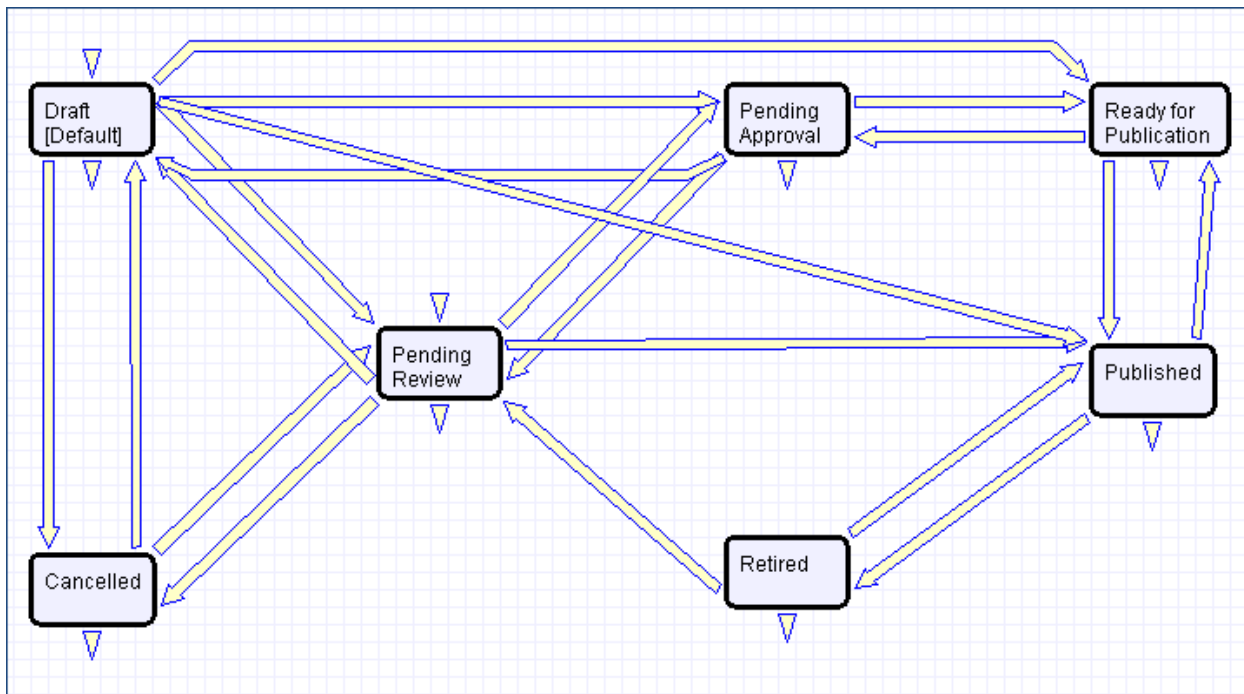
Project Management Tables

Projects Table

Purpose: This table holds records for project management activities. It is currently optimized for companies providing consulting services to their clients, and allows them to manage their billable and unbillable hours, work authorizations, and overall project status. It can easily be customized for internal project management instead.

Use Case

Members of the Admin or Project Manager groups may manually create Project records. Projects are creatable only in the "Planned", "Assigned", "Work in Progress" and "Awaiting Customer Feedback" states.

Only members of the Product Manager and admin Groups may edit others' Project records, but Support Staff (Base ServiceDesk group) may view their own Project records. The user who created the Project is automatically set as the Project Manager, and will get email notifications pertaining to the Project's status, such as when all tasks are completed or the hours spent on the project have exceeded what was authorized. Project CCs can be specified who will also receive notice when the Project is completed. The information on the "Contact Information" tab will be filled out with the information of the Project Manager's Manager, provided by the Employee record of the Project Manager.

Project records are divided into two broad categories: Internal and Client-Related. These categories are further divided by type. The tasks that are automatically generated by a project depend on the type selected. For each Project type there are task templates that list the tasks that get automatically generated. The exact selection of tasks that will be generated can be specified on the "Tasks" tab of the Project record. Ad-hoc tasks can also be created using an action button on that tab. Additional fields are visible when a Project's Category is "Client-Related". These fields hold information about customer contracts, contact details, and authorized hours.

Project

Save ▾	Cancel ▾	**Project Details**	Contact Information	Time / Billing	Tasks	Emails	History

Project Manager   Roger Winston 🔍   Status   [Work In Progress ▾]   ID   101

Project Category   Client-Related   *Project Type   [Client Professional Services ▾]

*Project Name   [Server Support]

Company   [Agiloft] 🔍

Project Details

Project CCs   Karen Sneller 🔍

Project Description   [Provide support for Agiloft's purchased servers.]

Internal Project Notes   [ ]

Estimated Hours   [150]

Contract Approved by Customer   ● Yes ○ No   Date Contract Approved   [Jan 17 2014] 📅 [00:00] 🕐

Next Milestone Date   [ ] 📅   Target Completion Date   [ ] 📅

Attached Documents

Attached Documents   [Attach/Manage]   Attached Contract   [Attach/Manage]
                      No Files Attached   No Files Attached

Time spent on the project is tracked on the "Time / Billing" tab. It includes a small form to enter time spent and a related table of time entries for this project. Time entries can be searched for in the related table and linked to the project manually.

Workflow
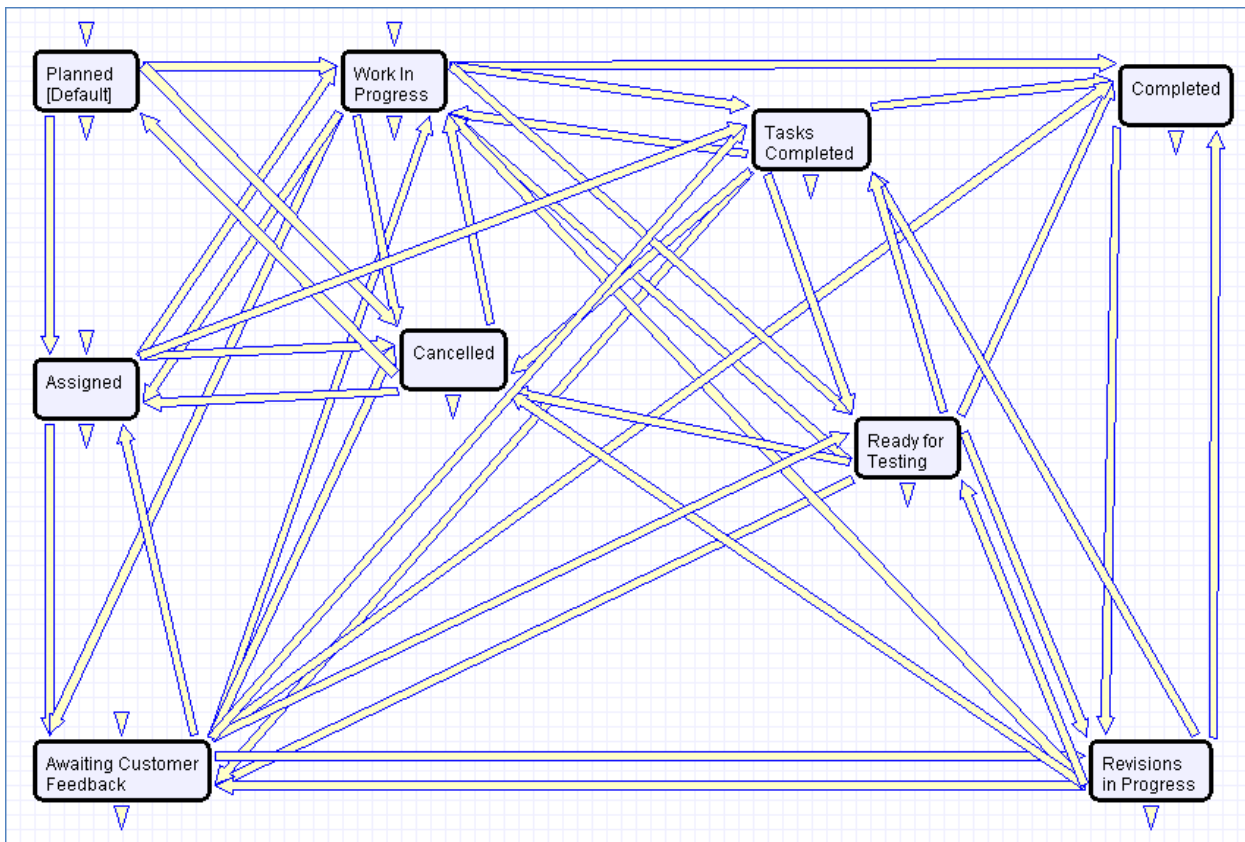
Ownership

Records in this table are "owned" by the Employee designated as the Project Manager.

Project Types Table

Purpose: This is a background table that holds the Project Types referenced by the Projects table. It allows the creation of new Project Types by Project Managers and other users without Admin group privileges.
Project Types define the task generation method, if any, and if task workflows will be used, they are defined in the project type record:

In addition to Predefined Task Workflows, project types may use Users Selected Tasks or User Generated Ad Hoc Tasks as the task generation methods.

The differences between these methods is discussed in more detail in the Tasks for Service Requests section above, and the same options available for Service Requests apply to Projects. The methodology for setting up task workflows and the way in which the prerequisite tasks are handled is discussed above in the Task Workflows Table section and in the Task Templates section.

Purchase Orders (PO) Table

Purpose: The Purchase Order table tracks authorized billable hours for a project. It is shown as a related table within the Project table. It could easily be linked to support cases or Quotes or other tables within the system.

Use Case

PO records are creatable manually via the web form, from within Project records in the related table, or via mass import. Only members of the Professional Services, Sales and admin Groups may create or import records. PO records are creatable in any workflow state.

Only members of the Professional Services, Sales and admin Groups may edit records, but Support Staff may view all PO records.
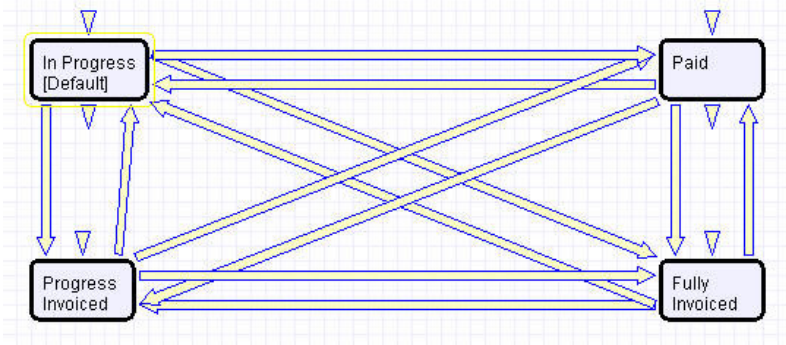
Ownership

Records in this table are "owned" by an Employee, generally the one who created the record. Each record is associated with a particular Employee login.

Support Cases Table

Purpose: This table is used to manage external customer support requests. It is set up to be accessed by the Customer group and the Base Technical Group.

End User Record Submission

Customers may create support cases using the tab in the end user interface or by sending an email (once an inbound email address is set up). When a customer submits a support case, the contact information fields automatically populate based on the details in his/her user record. If the user record doesn't contain a value in the Customer Name or Email fields, the customer will be required to enter a value in those fields manually. The Type of Issue is set by default to Question. If the customer changes it to Installation Issue or Bug, they will be required to fill out the Steps to Reproduce field. The case is assigned by default to the Support Team and the default status is Open.

Technician Record Submission

Staff members may also submit support cases on behalf of a customer, associating the customer with the case. If an internal staff user creates a case, he may assign it directly to an individual or a team other than the Support Team and may set its starting status to Open, Assigned, or Closed. When the record is created, emails are sent to the customer acknowledging receipt of the support case and to the assigned team (or person) telling them the case has been assigned to them.

If a Support Staff technician creates a record in a status of Closed (because they spoke with the customer over the phone and resolved the issue), an email is sent to the customer telling them how to reopen their case. Workflow actions send these emails automatically, but staff users can override them if given permission to override workflow actions.

Additional Processing of Records

When a technician works on a case, if he needs more information from the customer in order to take further action, he can set the status to Sent to Customer. This will automatically send an email to the customer requesting further information and include the content of the Additional Notes field, an append-only field that is used to communicate with the customer. The email includes a hyperlink for the customer to click to login to edit the case directly.

When the customer edits the case or replies to the email, the status changes to Updated by Customer and an email notifies the assigned person that the customer has replied. The customer is able to update the Additional Notes field directly and any text from an email reply maps to that same field. If the customer updates the case at any point, an email notifies the assigned person of the update. If the technician needs to reassign the case to someone else, he or she simply changes the Assigned Person field to that person's name and the system will email the new assignee notifying them of the reassignment.

The Staff Only Notes field holds working notes that should not be visible to the customer.

When the technician has completed work on the case, he/she sets the Status field to Closed and puts the

solution notes into the Solution field. This triggers an email to the customer that includes the content of the Solution field and tells the customer that the work is done. This closing email gives the customer a hotlink back to the record if they wish to reopen it and instructs them to explain why they are not satisfied with the solution. Clicking the hotlink will automatically change the "I Would Like To Reopen My Ticket" field to Yes, which in turn sets the Status of the ticket to Reopened and notifies the assigned person.

By default no escalation rules are set up for the Support Case table.
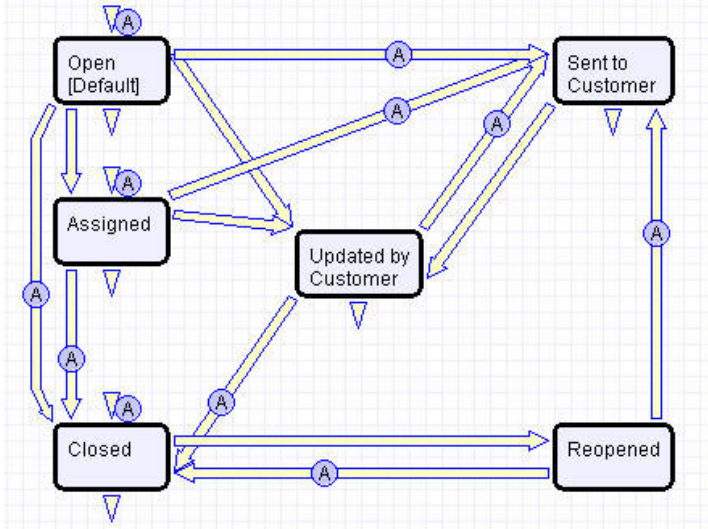
Ownership

Records in this table are "owned" by the individual customer. This means each record is associated with a particular customer login and no other customers will be able to edit that record, though members of the Customer Manager group can view all records submitted by other people at their company.

SalesCRM Tables - Leads Table

Purpose: This table is used as the initial point of entry for sales leads. Leads may self-register at your website or may be imported from a lead generation program. The initial qualification is done in this table, which contains all the fields desirable for managing sales.

Once a lead is qualified, it may either be fully worked in this table, or converted into a contact, an opportunity, and a company/account record, and the sales process may be managed in those records.

Record Creation

Leads can be created directly using the web form, or, once an inbound email address is set up, via email. New leads may only start out as Qualified or Unqualified in the workflow. The Leads table is set up by default to allow those in the Guest group to create records, allowing "click to register" lead generation hotlinks, and the embedding of the lead creation form in a web page. Naturally leads may also be imported from a spreadsheet from a lead generation program.

Processing of Records

Admins and members of the Sales group can create, view, and edit Leads. No other groups have access to the table by default.

When a lead's status changes to "Converted", Agiloft converts the information in the record into three new records in the three other tables: Company, Opportunity, and Contact. This order of creation is important because both the Opportunity and Contact records will contain links to the original Company via the new record. If the Company record is not created first, the Contact and Opportunity records will be unlinked and orphaned, disabling reporting features.

Data fields containing information relevant to the company, such as address and billing address, company website, industry, annual revenue and number of employees, map to the Company record. Sales-specific data fields, such as key requirements, earliest and latest possible close date, and sales actions taken, map to the Opportunity record. All data from the lead referencing a specific person at the company map into a new Contact containing the individual's desk and cell phone numbers, email address, email preferences, work hours, and so on.
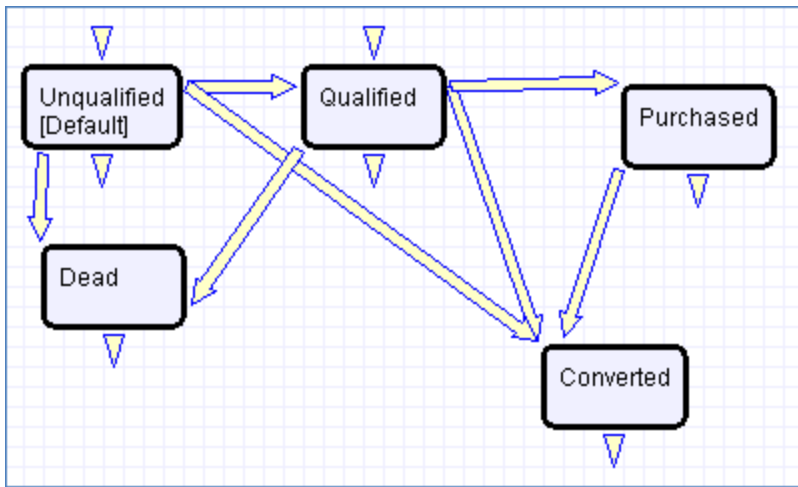
Ownership

Records in this table are "owned" by the individual assigned sales rep, so each record is associated with a particular user login. Only members of the Sales and admin groups can view or edit Leads.

Sales CRM Tables Opportunities/Deals Tables

Purpose: This table tracks sales opportunity information and contract information when an opportunity becomes a sale. An opportunity, when closed, is simply changed into a Deal, and all information is maintained, while new contract fields appear.

Most Opportunity records are created from Leads via a Conversion rule. Deals are typically created from Opportunities when they change to "Closed Won". Members of the Admin or Sales groups may create records manually. An Opportunity record should be considered a possible sale, while a Deal is an actual finalized and confirmed agreement.

Admins and members of the Sales group can create, view and edit Opportunities. Members of the Professional Services and Support Staff groups may view all Opportunities. No other groups have access to the table by default. The Opportunities table tracks information on possible sales for your organization. The Deals section handles Opportunities that have been successfully closed and turned into contracts and contains specialized fields for each contract's unique information, such as SLA requirements, renewal dates, and so on. When an Opportunity's status changes to "Closed Won", a rule converts it to a new Deal record.
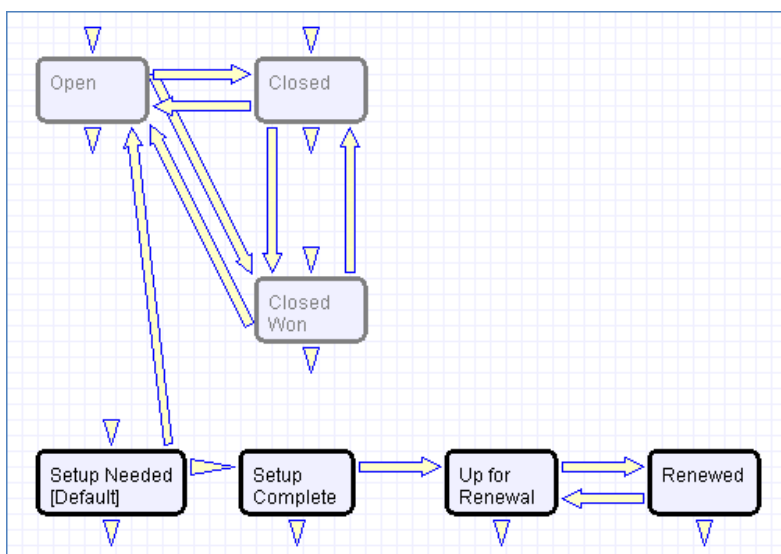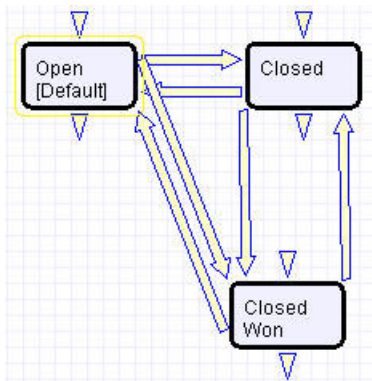
Ownership

Records in this table are "owned" by the individual assigned sales rep, so each record is associated with a particular user login from the Contacts table.

Appendix A – Detailed Rules Setup

In this Appendix we provide detailed technical information on the rules and actions that are preconfigured and other details of the out-of-the-box implementation. The rules and actions that run in the background to validate that users follow the correct processes and to automate process steps and notifications are documented with screenshots and details.

This automation is described in more general terms in the Use Case section of the main document for each table.

The tables are listed alphabetically here.

- Approvals Rules
- Approval Templates Table Rules
- Approval Workflows Table Rules
- Change Requests Table Rules
- Companies Table Rules
- Configuration Items Table Rules
- Contracts Table Rules
- Documents Table Rules
- Incidents Table Rules
- Insurance Certificates Table Rules
- Items Requested Table Rules
- Leads Table Rules
- People Table Rules
- Problems Table Rules
- Projects Table Rules
- Project Types Table Rules
- Purchase Orders Table Rules
- Purchase Requests Table Rules
- Service Requests Table Rules
- Support Cases Table Rules
- Tasks Table Rules
- Task Templates Table Rules

Approvals Rules

The Approvals table has the following rules set up.

Create: All Contract Approval Create Actions (web, API)

Purpose: This handles the creation and assignment actions for all newly created contract approval records.

Runs: When created by web or API.

Search Filter: Contract ID is not empty

Priority: 1

Actions:

- I: Contract Create and Assign Actions

```
Set Related To to Contracts
Prepend Contract Title to Approval Title
Set Approval Packet Files
Set Next Step
Set Concurrent Approvals
Set Lowest Step Number
Set Next Approvals in approval record
if (Assign Approval Based On='Person from Contract') {
    Set Approver Based on Variable
} else if (Assign Approval Based On='Team from Contract') {
    Set Approval Team Based on Variable
}
```

- I: Set Approval Team if empty

```
if (Approval Team=NULL or Assign Approval Based On='Person from Contract') {
    if (Approver!=NULL) {
        Set Approval Team to Primary Team of Approver
    } else {
        Set Approval Team to Contract Assigned Team
    }
}
```

Create: Document Approval Creation Actions (web, API)

Purpose: This handles the creation, assignment, and notification actions for all newly created document approval records.

Runs: When created by web or API

Search filter: Document ID is not empty

Priority: 1

Actions:

■ I: Create Actions for Document Approvals

```
Set Related To to Documents
Set Approval Team to Doc Reviewers Team and Status to Pending Approval
Email Document Reviewer of Pending Approval
Set Document Approval Title
```

Create: All Change Approval Creation Actions (web, API) – New

Purpose: This handles the setting of concurrent approvals, next step, lowest step number and all next approvals as well as the assignment actions

Runs: When created by web or API

Search filter: R: Change Request ID is Not Null

I: CR Creation Actions:

- U: Set Next Step
- U: Set Concurrent Approvals
- U: Set Lowest Step Number
- U: Set Next Approvals in approval record
- if (Assign Approval Based On='Person from Change Request')
- U: Set CR Approver based on Variable
- Elseif (Assign Approval Based On=Team from Change Request)
- U: Set CR Approval Team based on Variable

I: Assign Actions

- if (Approval Team equals NULL)
- if (Approver does not equal NULL)
- U: Set Approval Team to Primary Team of Approver

Create: Notify if Needed for Contract Approval (web, API)

Purpose: If the approval is created in a status of Pending Approval, notify the Approver or Approval Team or auto-approve if appropriate. This would apply to an ad hoc approval when it is created.

Runs: When created by web or API

Priority: 3

Search Filter: Status is Pending Approval and Contract ID is not empty (i.e., it is for a contract)

I: Handle Approvals created in Pending Approval Status

```
if (Auto-Approve!='Yes') {
    if (Approver!='$null') {
        Email Contract Approver for Pending Approval
    } else {
        Email Contract Approval Team of Pending Approval
    }
} else {
    Email Contract Approval Team auto-approve notification
    Increment Number of Approvals Completed
    if (Number of Open Concurrent Approvals=0) {
        Set Next Approvals to Pending Approval Status
    }
}
```

Edit: All Edit Actions for Document Approval Updates (web and API)

Purpose: This handles all document-related approvals and rejections, as well as approval notes updates.

Runs: When edited by web or API

Search filter: Related to Documents

Priority: 4

I: All Document Approval Edit Actions

```
if (Approval Notes changed from: any value to: any value during record's last modification) {
    Copy Approval Notes to Document
    Append Approval Notes to Running Approval Notes
    Blank Out Approval Notes
}
if (Status changed from: any value to: 'Requires Change' during record's last modification) {
    Notify other approvers of required change
    Email Document Manager that doc requires changes
} else if (Status changed from: any value to: 'Permanently Rejected' during record's last modification) {
    Notify other approvers of permanent rejection for document
    Set Status of Document to Canceled
} else if (Status changed from: 'Requires Change' to: 'Pending Approval' during record's last modification) {
    Email Document Approver that document requires reapproval
    Blank approval notes
    Blank Approval Date and Approver
} else if (Status changed from: any value to: 'Approved' during record's last modification) {
    Update Number of Approvals Needed in Document
}
```

Edit: All Contract Approval Edit Actions

Purpose: Handles updates to approvals for a contract. Launches the next approval in the workflow when an approval is completed.

Runs: When edited by web or API

Priority: 5

Saved Search: Related to Contracts

I: All Contract Approval Edit Actions

```
if (Approval Notes changed from: any value to: any value last user's modification) {
    Copy approval notes to contract
    Append Approval Notes to Running Approval Notes
    Blank Out Approval Notes
}
if (Status changed from: any value to: 'Pending Approval' during record's last modification) {
    Increment Approval Round
    if (Auto-Approve='Yes') {
        Set Status to Approved
        Email Contract Approval Team auto-approve notification
    } else {
        if (Approver ID!=NULL) {
            Email Contract Approver for Pending Approval
        } else {
            Email Contract Approval Team of Pending Approval
        }
    }
} else if (Status changed from: any value to: 'Permanently Rejected' during record's last modification) {
    Email Contract Manager of rejection
    Set Queued Approvals to Not Needed
    Set contract Status to Canceled
    Append Rejected to Contract Updates
} else if (Status changed from: 'Pending Approval' to: 'Not Needed' during record's last modification) {
    if (Approver ID!=NULL) {
        Email Contract Approver that pending approval is no longer needed
    } else
    {
        Email Contract Approval Team that pending approval is no longer needed
    }
} else if (Status changed from: any value to: 'Requires Change' during record's last modification) {
    Email Contract Owner that an approver has requested a change
    Set related approvals to Queued if not Queued or Not Needed
    Set contract's Status to Pending Contract Manager
} else if (Status changed from: 'Pending Approval' to: 'Queued' during record's last modification) {
    if (Approver ID!=NULL) {
        Email Contract Approver that approval has been set to Queued
    } else
    {
        Email Contract Approval Team that approval has been set to Queued
    }
} else if (Status changed from: any value to: 'Approved' during record's last modification) {
    Increment Number of Approvals in Contract
    if (Number of Open Concurrent Approvals=0) {
        Set Next Approvals to Pending Approval Status
    }
    if (Approver!=Updated By) {
        Set Approver to Last Updater
    }
}
```

Edit: Approval by Email (web)

Purpose: Checks for updates by email that change the Approved by Email field from No to Yes, and changes the Status to Approved accordingly
Runs: On edit by web
Priority: 6
Saved Search: R: Approved by Email changed to Yes and Status is Pending Approval

- U: Set Approved by Email to No

■ U: Set Approval Status to Approved

Edit: All Change Approval Edit Actions (web, API) – New

Purpose: This handles the updates of CR for approval notes and workflow for approvals, rejections or changes.
Runs: When edited by web or API
Condition: When an Approval is edited by means of Web or API
Saved Search: R: Related to Change Request

```
if (Approval Notes changed from: any value to: any value during record's last modification) {
    Copy Approval Notes to Change Request Approval Notes
    Append Approval Notes to Running Approval Notes
    Blank Out Approval Notes
}
if (Status changed from: any value to: 'Pending Approval' during record's last modification) {
    Set Next Step
    Set Concurrent Approvals
    Set Next Approvals in approval record
    Increment Approval Round
    if (Auto-Approve='Yes') {
        Set Status to Approved
        if (Notify for Auto-Approval='Yes') {
            Email Change Approval Team auto-approve notification
        }
    } else {
        if (Approver ID!=NULL) {
            Email Change Approver of Pending Approval
        } else {
            Email Change Approval Team of Pending Approval
        }
    }
    if (Approved/Rejected By!=NULL) {
        Blank Approval Fields
    }
    if (Change Request Status='Pending Change Manager') {
        Set Change Status to Pending Approval
    }
} else if (Status changed from: any value to: 'Approved' during record's last modification) {
    if (Number of Open Concurrent Approvals=0) {
        Set Next Approvals to Pending Approval Status
    }
    if (Approver!=Updated By) {
        Set Approver to Last Updater
    }
    Refresh Completed Approvals in CR
} else if (Status changed from: any value to: 'Requires Change' during record's last modification) {
    Email Change Mgmt Team and Change Mgr Changes Required
    Set Change Status to Pending Change Manager
} else if (Status changed from: any value to: 'Permanently Rejected' during record's last modification) {
    Set Queued and Pending Change Approvals to Not Needed
    Set Change Status to Rejected
    Email Change Requester and Chang Mgmt Team of Rejection
} else if (Status changed from: 'Pending Approval' to: 'Not Needed' during record's last modification) {
    if (Approver ID!=NULL) {
        Email Change Approver that Approval is no longer needed
    }
    Email Change Approval Team that approval no longer needed
} else if (Status changed from: 'Pending Approval' to: 'Queued' during record's last modification) {
    if (Approver ID!=NULL) {
        Email Change Approver that approval is Queued
    } else
    {
        Email Change Approval team that approval is Queued
```

Approval Templates Table Rules

The Approval Templates table has the following rules set up.

Create/Edit: Blank out fields depending on Assign Approval Based On (web, API)

Purpose: Clear out the non-applicable fields depending on the value of Assign Approval Based On.

Runs: When edited by web or API

Search Filter: None

Priority: 1

- I: Set Approval Template Relationship

```
if (Workflow Related to='Contracts' and Related To=NULL) {
        Set Related To Contracts
} else if (Workflow Related to='Change Request' and Related To=NULL) {
        Set Related To Change Requests
}
```

- I: Blank out fields based on Assign Approval Based On

```
if (Assign Approval Based On='Fields in Approval Template') {
        Blank out Assign To
} else if (Assign Approval Based On='Person from Contract' or Assign Approval Based On='Person from Change Request' or Assign Approval Based On='Team from Contract' or Assign Approval Based On='Team from Change Request') {
        Blank out Approval Team and Approver
}
```

Edit: All Contract Approval Edit Actions (web, API)

Purpose: This rule handles the creation of an approval record from an approval template when the relevant workflow is selected.

Runs: When edited by web or API.

Search Filter: Workflow Related To is Contracts

Priority: 2

Actions:

- I: Handle Flags to Generate or Check Approvals

```
if (Trigger Approval Creation='Yes') {
    if (Generated for Contract IDs!~==Latest Contract ID and Latest Contract ID!=NULL and (Approval
    Usage='Required' or Approval Usage='Conditional' and Condition='TRUE')) {
        Convert to Approval
        Update Generated for Contract IDs
    }
    Set Trigger Approval Creation to No
} else if (Flag to Clone Template='Yes') {
    Convert to Approval Template
    Set Flag to Clone Template to No
} else if (Flag to Recheck Approval='Yes') {
    if (Generated for Contract IDs!~==Latest Contract ID and Condition='TRUE') {
        Convert to Approval
        Update Generated for Contract IDs
    } else if (Generated for Contract IDs~==Latest Contract ID and Condition='FALSE') {
        Set unnecessary Approval to Not Needed
        Conditional Approval is no longer needed
    } else if (Generated for Contract IDs~==Latest Contract ID and Condition='TRUE') {
        Set Status of Not Needed approval to Queued
    }
    Set Flag to Recheck Approval to No
}
```

All Change Approval Edit Actions (web, API)

Purpose: This handles the conversion of Change Request approvals, the checks for conditional approvals.

Runs: When edited by web or API

Search filter: workflow related to Change Requests

Priority: 2

I: Handle Flags to Generate or Check CR Approvals

- if (Trigger Approval Creation='Yes') {
- if (Generated for Change Request IDs does not contain item Latest Change Request ID and Latest Change Request ID does not equal NULL and (Approval Usage='Required' or Approval Usage='Conditional' and Condition='TRUE'))
- C: Convert to CR Approval
- U: Update Generated for CR IDs
- U: Set Trigger Approval Creation to No
- else if (Flag to Clone Template='Yes')
- C: Convert to Approval Template
- U: Set Flag to Clone Template to No
- else if (Flag to Recheck Approval='Yes')
- U: Set Flag to Recheck Approval to No
- if (Generated for Change Request IDs~==Latest Change Request ID and Condition='FALSE')
- B: Conditional Approval is no longer needed
- L: Set unnecessary CR approval to not needed
- else if (Generated for Change Request IDs~==Latest Change Request ID and Condition='TRUE')
- L: Set Not Needed CR approval to Queued (filtered to only those whose current status is not needed)
- else if (Generated for Change Request IDs!~==Latest Change Request ID and Condition='TRUE')
- C: Convert to CR Approval

- U: Update Generated for CR IDs

Approval Workflows Table Rules

Rules and Actions:

Create: All Create Actions (web, API)

Purpose: This rule handles workflows created by cloning.

Runs: When created by web or API

Search filter: None

Priority: 1

I: All Create Actions

```
if (Source Workflow ID!=NULL) {
        Map ID into Source Workflow Approval Templates
        Set Cloned Workflow Title
}
```

Change Requests Table Rules

Rules and actions:

Rule: Change Request – All Creation actions

Runs when created by any method, no saved search.

- I: Validations and Conditional Actions

```
if (Spawning SR ID!=NULL) {
      Update "Additional Notes" in Spawning SR
}
if (Spawning Problem ID!=NULL) {
      Update "Additional Notes" in Spawning Problem
}
if (Parent CR ID!=NULL) {
      Set CR Type to Child
}
if (Related to='Configuration Item' and CI Name=NULL) {
      Must select a CI
}
```

- I: Assign Change Request

```
if (Give Responsible Team Priority over CI Team!='Yes' and Assigned Team=NULL and CI Responsible Team!=NULL) {
      Set Assigned Team to CI Responsible Team
} else if (Assigned Team=NULL and Service Responsible Team!=NULL) {
      Set Assigned Team to Service Responsible Team
} else {
      Set Assigned Team to 1st Level Support Team
}
```

- I: Send Emails to Requestor and Assignee

```
if (Status!='Closed') {
      Send Acknowledgment Email to Customer
      if (Assigned Person!=NULL) {
            Send New CR email to Assigned Person
      } else {
            Send New CR email to Assigned Team
      }
}
```

Rule: CR – Approvals

Runs when created or edited, with API, Email, or Web, no search

- I: Approver Validations

```
if (Approved by 1st Approver changed from: any value to: any of 'No','Yes' last user's modification and 1st Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 1st Approver
}
if (Approved by 2nd Approver changed from: any value to: any of 'No','Yes' last user's modification and 2nd Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 2nd Approver
}
if (Approved by 3rd Approver changed from: any value to: any of 'No','Yes' last user's modification and 3rd Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 3rd Approver
}
if (Approved by 4th Approver changed from: any value to: any of 'No','Yes' last user's modification and 4th Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 4th Approver
}
if (Approved by 5th Approver changed from: any value to: any of 'No','Yes' last user's modification and 5th Approval
Team!<<'Company Team,1st Level Support Team,Admin Team,Change Approver Team,Document Reviewers Team') {
     Must be 5th Approver
}
```

Note: Matches against `$GLOBAL.my_teams` variable.

- I: Status Change Validations

```
if (Status changed from: any value to: 'Approved' last user's modification) {
     if (Approved by 1st Approver!='Yes') {
          Need all approvals
     }
     if (Number of Approvals Needed>=2 and Approved by 2nd Approver!='Yes') {
          Need all approvals
     }
     if (Number of Approvals Needed>=3 and Approved by 3rd Approver!='Yes') {
          Need all approvals
     }
     if (Number of Approvals Needed>=4 and Approved by 4th Approver!='Yes') {
          Need all approvals
     }
     if (Number of Approvals Needed>=5 and Approved by 5th Approver!='Yes') {
          Need all approvals
     }
}
```

- I: Approval Notification Actions

```
if (Status changed from: any value to: 'Pending Approval' last user's modification) {
    if (1st Approver!=NULL) {
        Send 1st Approver email
    } else if (1st Approval Team!=NULL) {
        Send 1st Approval Team email
    }
    if (2nd Approver!=NULL) {
        Email 2nd approver need approval
    } else if (2nd Approval Team!=NULL) {
        Email second approval team need approval
    }
    if (3rd Approver!=NULL) {
        Email 3rd approver need approval
    } else if (3rd Approval Team!=NULL) {
        Email 3rd approval team need approval
    }
    if (4th Approver!=NULL) {
        Email 4th approver need approval
    } else if (4th Approval Team!=NULL) {
        Email 4th approval team need approval
    }
    if (5th Approver!=NULL) {
        Email 5th approver need approval
    } else if (5th Approval Team!=NULL) {
        Email 5th approval team need approval
    }
}
if (Status changed from: any value to: 'Approved' during record's last modification and Updated By!='$assigned_person') {
    if (Assigned Person!=NULL) {
        Send CR Approved email to Assigned Person
    } else if (Assigned Team!=NULL) {
        Send CR Approved email to Assigned Team
    }
} else if (Status changed from: any value to: 'Rejected' last user's modification and Updated By!='$assigned_person') {
    if (Assigned Person!=NULL) {
        Send CR Rejected email to Assigned Person
    } else if (Assigned Team!=NULL) {
        Send CR Rejected email to Assigned Team
    }
}
```

Rule: Change Request - Edit actions (API Enabled)

- ■ I: Related record Actions

```
if (Status changed from: any value to: 'Closed' during record's last modification) {
    if (Spawning Problem ID!=NULL) {
        Set "CR Closed?" in Problem to Yes
    }
    if (Spawning SR ID!=NULL) {
        Set "CR Closed?" in Service Request to Yes
    }
}
```

- ■ I: Assignee Change Notifications

```
if (Updated By='$assigned_person' and (Assigned Person changed from: any value to: any value last user's modification or
Assigned Team changed from: any value to: any value last user's modification)) {
    if (Assigned Person!=NULL) {
        Email Assigned Person of new assignment
    } else {
        Email Assigned Team of new assignment
    }
    if (Assigned Person changed from: any value to: any value last user's modification) {
        Increment Number of Assignees
    }
    if (Assigned Team changed from: any value to: any value last user's modification) {
        Increment Number of Teams Assigned
    }
}
```

Rule: Parent/Child CR Handling

Runs on create and edit, including edits by other rules on creation

Saved Search: CR Type != Ind.

```
if (CR Type='Parent') {
    if (Status changed from: any value to: 'Approved' during record's last modification) {
        Approve linked Child CRs
    } else if (Status changed from: any value to: 'Rejected' during record's last modification) {
        Set status of child CRs to Rejected
    }
    if (CIs for Child CRs changed from: any value to: any value last user's modification) {
        Set Latest CR Field in Linked CI
    }
}
if (CR Type='Child' and Change Summary!~='child of') {
    Add Child of prefix to change summary
}
```

Action Buttons

The action buttons for Change Requests are listed below:

Add Time

Purpose: Provide users with a way to add Time Entries to a Change Request from the Change Request record itself.

On Click: Save Record, then execute action.

Actions:

- I: Add Time

```
if (Time Spent=NULL) {
    Time spent must not be blank
}
Convert to Time Entry
Set Time Spent and Time Description to Blank
```

Notes: Triggers action to validate that Time Spent is not blank, then converts Time Spent and Time Description fields to a Time Entry record linked to the current Change Request record, then blanks out Time Spent and Time Description

Set CI to Active

Purpose: Updates the linked CI to change its status to Active

On Click: Execute action

Actions:

- U: Set Status of linked CI to Active

Update Field: Configuration_Item:Status

Update Value: Installed

Set CI to In Maintenance

Purpose: Updates the linked CI to change its Status to In Maintenance

On Click: Execute Action

Actions:

- U: Set linked CI to In Maintenance

Update Field: Configuration_Item:Status

Update Value: In Maintenance

Spawn Child CR

Purpose: Converts the CR to multiple child CRs linked to the CI's selected in the CIs for Child CRs field.

On Click: Execute action

Options: Visible only when CR Type is Parent

Actions:

- ■ C: Spawn Child CR

Conversion: Spawn Child CR

Options: Allow user to reject creation, create record if does not exist, conversion runs interactively

Companies Table Rules

The Companies table saved searches and action buttons are defined below:

Saved Searches

The default searches provided for this table are detailed below.

Saved Search Name	Search Description
My Company	Returns records where the Company Name and Company City match those of the logged-in user
Reference Accounts	Shows all records for companies who have agreed to
Customers	Company Roles includes "Customer"
Manufacturers	Company Roles includes "Manufacturer"
Partners	Company Roles includes "Partner"
Prospects	Company Roles includes "Prospect"
Vendors	Company Roles includes "Vendor"

Action Buttons

Find local hotels	
Purpose	Automatically searches Google Maps for hotels near the company.
On Click	Opens URL: "http://maps.google.com/maps?f=q&q=hotels" + "loc:" + urlEncode($street_address_1) + "," + urlEncode($city) + "," + $us_state

Map it	
Purpose	Automatically searches Google Maps for the location of the company.
On Click	Opens URL: "http://maps.google.com/maps?f=q&q=" + urlEncode($street_address_1) + "," + urlEncode($city) + "," + $us_state

Refresh User	
Purpose	Saves record and reopens for edit to set most recent user to person editing record.
On Click	Save and Open record for edit

Configuration Items Table Rules

The Configuration Items Table list of Configuration Items, rules and actions is described below:

Configuration Item Type and Subtype Dependencies

The tables below show the current setup of dependencies between CI Type and CI Subtype. These dependencies can be changed, as can all the choices, to meet your needs. The dependencies are also shown in the spreadsheet fields_and_choices.xsl.

CI Type values	CI Subtype (linked) values
Communication Device	Aircard Conference Phone Desk Phone Fax Machine IP Phone Mobile Phone Modem PDA enabled Cell Phone Video Conferencing Voice Gateway WiFi
Computer	Desktop Laptop Tablet Virtual Machine Windows Workstation
Computer Peripheral	Bar Code Scanner Camera Disk Drive Docking Station External Hard Drive Modem Monitor Mouse Peripheral - Other UPS Video Card

Data Center	Modem Other Rack Router Server Storage Switch UPS Virtual Machine Wireless Access Point
Imaging/Printing	Copier Fax Machine Label Maker Network Printer Personal Printer Scanner
Infrastructure Item	Cluster Firewall FTP Server Load Balancer Monitoring Device Rack Video Conferencing Virtual Machine Virtual Private Network Voice Gateway VPN Appliance Web Service Web Site
Network Hardware	IP Device IP Network IP Router IP Switch KVM Switch Load Balancer Monitoring Device Network Adapter Network Appliance Other Hardware Switch Virtual Machine Voice Gateway VPN Appliance WAN Interface

Server	Directory Server Email Server Firewall FTP Server Java Server Server Storage Management Server Virtual Machine
Storage Device	Storage Storage Device Storage Management Server Storage Switch Tape Library
Software Application	Application Server Database Application Database Instance Desktop Application Google Search Appliance Infrastructure Software Operating System Other Application Web Application Web Server Application Web Service Windows Service
Storage Device	Storage Storage Device Storage Management Server Storage Switch Tape Library
Video Conferencing	Camera Other (DVD, Mixer, Amp) Projector Screen Television Video Conference Controller Web Camera
Other	
None	

Rules and Actions

Edit: Push Information to Task Template to Convert for Change Request
Purpose: Creates tasks based on workflow or templates.

Runs: Edit by Web or API

Search Filter: R: Flag to Create Task from Workflow or Template is Yes

Priority: 5

Actions:

- I: Trigger Task Conversion within Task Template

```
if (Flag to Create Tasks from Workflow='Yes') {
        Set Flag to Create Tasks from Workflow to No
        Update Task Templates through Workflow for Task Conversion
} else if (Flag to Create Tasks from Template='Yes') {
        Set Flag to Create Tasks from Template to No
        Update Task Template directly for task conversion
}
```

Edit: All Edit Actions

Purpose: Adds assets to Contracts when Latest Contract ID added to.

Runs: Edit by Web or API

Search Filter: None

Priority: 11

Actions:

- I: Edit actions by API or web

```
if (Latest Contract ID changed from: any value to: any value during record's last modification) {
        Update linked Contract to add this asset
}
```

Contracts Table Rules

The Contracts table has the following rules set up. Rules which are run based on a schedule (rather than those which are event-triggered) are identified by "TB", for "Time-based".

Create: All New Contract Actions

Purpose: This rule handles all update and linking actions upon creating a new record.

Runs: Upon creation by Web or API

Search Filter: None

Priority: 1

Actions:

- I: All New Contract Actions

```
if (Contract Start Date!=NULL and Contract End Date!=NULL and Contract End Date<Contract Start Date) {
    contract end date can't be less than start date
}
if (Status='Signed' and Contract End Date>1 day in the future and Contract Start Date>1 minute old) {
    Set Status to Active
}
if (Renewal Contract ID!=NULL and Renewal Contract='No') {
    Blank out Previous Contract ID
}
if (Previous Contract ID!=NULL and Renewal Contract='Yes') {
    if (CI Involvement='For One or More CI's') {
        Copy contract id to latest contract id in assets
    }
    Copy contract id to previous contract Renewal Contract Id
}
if (Company ID!=NULL) {
    Set Default Contracts in Ins Certs for Contract Company
}
```

- I: Set Renewal Notification Date based on renewal type

```
if (Renewal Type='Auto-Renewing' and Advance Notice Required to Cancel Autorenewal (days)!=NULL) {
    Set notification date for autorenewing contract
} else if (Contract End Date!=NULL) {
    Set Renewal Notification Date
}
```

Updated Field: Renewal Notification Date

Updated Value: $contract_end_date – ($days_in_advance_to_notify_for_renewal) DAYS

Edit: Updates by Party

Purpose: This rule handles notifications for Contracts that are updated by an outside party that does not have direct access to the system.

Runs: Upon edits by Email or Web

Search Filter: R: Updates by Vendor or Customer Team

Priority: 1

Actions:

- E: Email Owner and Sender of Outside Party Update
    - **Sent To:** Contract Manager Team, Contract Manager, Sent to Party By.
    - **Summary:** This email notifies the relevant users that an outside party has updated the contract.
- I: Handle Inbound Files

```
if (Transitional Contract Files changed from: any value to: any value last user's modification and Transitional Contract Files!='$null') {
    Set Attachment Type and Title for Outside Party Update
}
```

Edit: All Edit Actions by Web or API

Purpose: This rule handles the processes resulting from changes made by staff or rules.

Runs: Upon edits by Web or API

Search Filter: None

Priority: 2

Actions:

- I: All Edit Actions with API

```
if (Approval Type='Parallel' and Number of Approvals Needed='$number_of_approvals_received' and Number of Approvals
Needed>=1 and Status='Pending Approval') {
        Set Status to Approved
        Email Contract Owner of approved contract
}
if (Status changed from: any value to: 'Pending Approval' during record's last modification and Approval Type='Sequential'
and Show Approval Steps='No') {
        Set Show Approval Fields to Yes
} else if (Status changed from: 'Pending Approval' to: any value last user's modification and Show Approval Steps='Yes') {
        Set Show Approval Steps to No
} else if (Status changed from: any value to: 'Signed' last user's modification and Contract End Date>1 day in the future and
Contract Start Date>=1 minute old) {
        Set Status to Active
} else if (Status changed from: 'Cancelled' to: 'Pending Contract Manager' last user's modification) {
        Set Status of existing approvals to Requires Reapproval
        Clear Approval Action
        if (Current Step ID!='$starting_step_id') {
                Set Current Step ID to Start Step ID
        }
}
if (Times Sent for Approval>=1 and Workflow Title changed from: any value to: any value last user's modification) {
        Not Allowed to Change Workflows
}
if (Contract End Date changed from: any value to: any value last user's modification or Days in advance to notify for renewal
changed last user's modification) {
        Update Renewal Notification Date
}
if (Number of Subcontracts>=1 and Has Related Subcontracts='No') {
        Set Has Subcontracts to Yes
}
if (Number of Related Contracts>=1 and Has Related Contracts='No') {
        Set Has Related Contracts to Yes
}
```

TB: Notify of upcoming expirations (Disabled by Default)

Purpose: This rule controls the email notifications for upcoming expirations of contracts

Runs: At selected time intervals; every 2 Days at 3 AM

Search Filter: R: Renewal date is tomorrow

Priority: 0

Action:

- E: Email Contract Owner about Renewal Date
    - **Sent To:** User in the Contract Manager field
    - **Summary:** Reminds the Contract Manager that the impending renewal date of a contract

TB: Daily Check for Start Date (Disabled by Default)

Purpose: This rule checks for contracts whose start date has arrived and changes the Status accordingly

Runs: At selected time intervals; every 2 days at 4 AM

Search Filter: R: Status is signed and Start Date is today or earlier

Priority: 4

- I: Actions When Start Date Arrives

```
Set Status to Active
if (Previous Contract ID!=NULL) {
      Set Status of Previous Contract to Renewed
}
```

TB: Daily Check for Expiration Date (disabled)

Purpose: This rule checks for contracts whose end date has arrived and takes the appropriate actions.

Runs: At selected time intervals; every 2 days at 2 AM

Search Filter: R: Contract Expired Today

Priority: 5

- I: Handle Expired Contract

```
if (Renewal Type='Automatically renews' and Status!='Renewed' and Status!='Cancelled') {
      Set Status to Renewed
      Convert to create new contract
} else {
      Set Status to Expired
}
if (Renewal Type='Notify staff to renew') {
      Email owner contract end date has passed
}
```

Action Buttons

The action buttons are listed below:

Cancel Contract

Purpose: This action button will cancel the contract and send an email to the contract requester. The button is only visible to Admins, Configuration Managers, Contract Owners, Service Managers, and Contract Managers.

On Click: Save record then execute actions.

Visibility: Only when Status is Active, Approved, Draft, Expired, Pending Approval, Renewed, or Signed.

Actions:

- I: Validations for Cancellations and Reapprovals

```
if (Approval Notes not changed last user's modification) {
      must add notes when rejecting
}
```

- ■ U: Set Status to Cancelled
  - ■ **Updated Field:** Status
  - ■ **Updated Value**: Cancelled
- ■ E: Email Requestor Contract Cancelled
  - ■ **Email Template:** Contract has been cancelled - Requester

Create and Attach Contract

Purpose: On click, when Contract is for Product Support and Upgrades, runs an attached record action to create and Support Contract template . If the Contract is for Subscription Service or Service Contract, runs an attached record action to attach an SLA template; Visible only when Contract is for a Service Contract, Subscription Service or Product Support and Upgrades; Pop-up description: Use this button to create and attach a contract based on fields in this record.

On Click: Execute action

Visibility: Only when Contract Type is Product Support and Upgrades, Service Contract, or Subscription Service.

- ■ I: Create contract based on type of contract

```
if (Contract Type='Product Support and Upgrades') {
      Attach Support Contract
} else if (Contract Type='Subscription Service' or Contract Type='Service Contract') {
      Attach SLA
}
```

Action Specific Items: Attach Support Contract and Attach SLA work as advertised.

Create Related Contract

Purpose: This button creates a new contract, mapping several of the fields and showing the new contract on the screen. When the new contract is saved, any assets linked to the old contract will be linked to the new one. Visible only when the Status is Renewed.

On Click: Save record, then execute actions.

Actions:

- ■ I: New Contract Validations

```
if (Record Type='Master Agreement' and New Record Type='Master Agreement' or Record Type='Contract' and New Record
Type='Contract' or Record Type='Subcontract' and New Record Type='Subcontract' or Record Type='Amendment' and New
Record Type='Amendment') {
      Cannot Create a New Record of this Type
}
```

- ■ C: Convert to new contract
  - ■ **Conversion:** Contract
  - ■ **Options:** Interactive, showing user new record page

Create Renewal Contract

Purpose: Validates that another renewal contract doesn't already exist, sets certain fields correctly, then converts the current contract to create a new contract, then sets Is the New Contract a Renewal to No.

On Click: Execute actions.

Visibility: Status is Active, Expired, or Signed

- I: Contract Renewal Validations

```
if (Renewal Contract ID!=NULL) {
        Cannot Create a Second Renewal
}
```

- I: Contract Renewal Actions

```
if (New Record Type!='$record_type') {
        Set New Record Type to Match Contract Record Type
}
if (Is the new contract a Renewal?='No') {
        Set Is the New Contract a Renewal to Yes
}
```

- C: Convert to create new contract
    - **Conversion:** Contract
    - **Options:** Interactive, showing user the new record page
- U: Set Is the New Contract a Renewal to No
    - **Updated Field:** Is the new contract a Renewal?
    - **Updated Value:** No

Mark As Signed

Purpose: Sets status to Active or Signed depending on if the Contract Start Date has passed or not.

On Click: Execute action.

Visibility: Status is Approved.

Actions:

- I: Contract Signing Actions

```
if (Contract Start Date<=this day) {
        Set Status to Active
} else {
        Set Status to Signed
}
```

Review Completed

Purpose: Depending on the Approval Action, the button either marks the Contract as approved, sets it to the next approval step, or back to the preceding or first step, and sends notifications to the contract owner (in case of rejection) or to the next assigned team otherwise.

On Click: Save record, then execute actions.

Visibility: Show Approval Steps is Yes.

Actions:

- ▪ I: Validations for review completed

```
if ((Approval Action='Rejected - Route Back One Step' or Approval Action='Rejected - Send Back for Resubmission') and
Approval Notes not changed last user's modification) {
       must add notes when rejecting
}
if (Current Approval Team!<<'Company Team,Internal Customer Team,Customer Team,Marketing Team,1st Level Support
Team,Sales Team,2nd Level Support Team,Admin Team,Professional Services Team,Change Management Team,Change
Approver Team,Configuration Management Team,Network Operations Team,Server Team,Custom Applications
Team,System Administration Team,Backup and Storage Team,Database Team,Security Team,Purchasing Team,Desktop
Applications Team,Contract Management Team,Vendor Management Team,HR Team,Facilities Team,Document Services
Team,Office Mgmt Team,Knowledge Team,Vendor Team,Legal Team,Risk Team,Compliance Team,Finance Team,Contract
Owner Team') {
       you are not on approval team
} else if (Status!='Pending Approval') {
       cannot approve while in draft status
}
```

  Note: Rule checking against $GLOBAL.my_teams

- ▪ C: Create Approval Record
  - ▪ **Conversion:** Approval
  - ▪ **Options:** Silently with no confirmation, don't report errors.

- I: Handle Review Complete Actions

```
if (Approval Action='NULL') {
    Must select an approval action
} else if (Approval Action='Approved - Route Forward') {
    if (Step Number='$total_number_of_steps') {
        Set Status to Approved
    } else {
        Set Next Step ID and clear approval action
        Email Current Approval Team ready for approval
    }
} else if (Approval Action='Rejected - Route Back One Step') {
    if (Previous Step ID=NULL) {
        Set Status back to Draft and clear approval action
        Email Assigned Team and Contract Owner of rejection
    } else {
        Set back to previous step and clear approval
        Email Current Assigned Team of Route Back
    }
} else if (Approval Action='Rejected - Send Back for Resubmission') {
    Set Status back to Draft and clear approval action
    Email Assigned Team and Contract Owner of rejection
    Set Status of existing approvals to Requires Reapproval
    if (Current Step ID!='$starting_step_id') {
        Set Current Step ID to Start Step ID
    }
}
```

Send Back for Reapproval

Purpose: Validates that notes are added, and then sets Step ID and Status to Start Step ID and Pending Approval or Requires Reapproval, as necessary. Emails Current Approval Team of reapproval.

On Click: Saves record then executes actions.

Visibility: Status is Approved.

Actions:

- I: Validations for Cancellations and Reapprovals

```
if (Approval Notes not changed last user's modification) {
    must add notes when rejecting
}
```

- I: Send Back for Reapproval Actions

```
if (Approval Type='Sequential') {
    Clear Approval Action
    if (Current Step ID!='$starting_step_id') {
        Set Current Step ID to Start Step ID
    }
    Email Current Approval Team of reapproval
    Set Status of existing approvals to Requires Reapproval
} else if (Approval Type='Parallel') {
    Set linked approvals to Pending Approval
}
Set Status to Pending Approval
```

Send Contract to Party

Purpose: Validates that notes are provided to Party then emails notes to party. If party email does not exist, generates a popup that warns the user that the email was not sent because there was no address.

On Click: Save record, execute actions, then save again.

Actions:

- I: Send to Party Actions

```
if (Notes To/From Party Contacts changed from: any value to: any value last user's modification and Company Main Contact
Email!=NULL) {
        Email Notes to Contract Party
        Set Sent to Party By field
} else if (Notes To/From Party Contacts not changed last user's modification) {
        Must provide content in Party Notes
}
if (Company Main Contact Email=NULL) {
        Warn that company email was blank
}
```

Submit for Approval

Purpose: On click, sets the User Status to Pending Approval then saves and closes the record; Visible only if Status equals Draft.

On Click: Saves record and executes action.

Visibility: Status is Draft or Pending Contract Manager.

Actions:

- I: Run Submit for Approval Actions

```
Set Status to Pending Approval
Increment Times Sent for Approval by 1
if (Approval Type='Sequential') {
        Set Show Approval Fields to Yes
        Set Current Step ID to Start Step ID
        Email Current Approval Team ready for approval
} else if (Approval Type='Parallel') {
        if (Times Sent for Approval=0) {
                Copy Contract ID to approval templates
        } else {
                Set Status of all approvals to Pending Approval
        }
}
```

Submit for Review

Purpose: Emails Contract Manager Team of New Contract, then sets Status to Pending Contract Manager.

On Click: Saves record, executes, actions, then saves record again.

Visibility: Status is Cancelled or Draft.

Actions:

- E: Email Contract Manager Team of New Contract
  - **Email Template:** Contract has been submitted for review.
- U: Set Status to Pending Contract Manager
  - **Updated Field:** Status
  - **Updated Value:** Pending Contract Manager

Documents Table Rules

Rules and Actions:

All Edit Validations

Purpose: This rule handles all validation actions when a Document is edited.

Runs: When edited by email or web.

Search Filter: None.

Priority: 5

Actions:

- I: All Edit Validations

```
if (Status='Pending Approval' and Requires Approval="Yes' and Number of Potential Reviewers<1) {
      Must Provide Potential Approvers
}
if (Total Number Still Awaiting Approval>=1 and Status changed from: any value to: any of 'Published','Ready for Publication' last user's modification) {
      if (Requires Approval="Yes') {
            requires approval before publishing
      } else {
            remind that they don't have approval yet
      }
}
```

Edit: Last Approval Received (API Enabled)

Purpose: This rule updates the Status and notifies the Submitter and Document Manager that the document is ready for publication.

Runs: When edited by email, web or API.

Search Filter: R: Final Approval just received

Priority: 5
Actions:

- U: Set Status to Ready for Publication
  - **Updated Field**: Status
  - **Updated Value:** Ready for Publication
- E: Email Submitter Document Ready for Publication
- E: Email Document Mgr that Document is Ready for Publication

Edit: All Edit Actions without API

Purpose: This rule handles all edit actions, including email notifications to Submitter and Document Manager of Status changes, and updates linked approval records.

Runs: When edited by email or web.

Search Filter: None.

Priority: 1

Actions:

- ◼ I: All Edit Actions

```
if (Status changed from: 'Pending Approval' to: 'Draft' last user's modification) {
    Email Submitter Document Requires Update
} else if (Status changed from: any value to: 'Published' last user's modification) {
    Email Submitter of Published Document
} else if (Status changed from: any value to: 'Pending Review' last user's modification) {
    Email Document Mgr that Document is Pending Review
} else if (Status changed from: any value to: 'Ready for Publication' last user's modification) {
    Email Submitter Document Ready for Publication
}
if (Reviewer(s) changed from: any value to: any value last user's modification and Reviewer(s)!=NULL and Status='Pending Approval') {
    Update Last Document ID in Employee
}
if (Status changed from: any of 'Draft','Pending Review' to: 'Pending Approval' last user's modification and (Total Number of Requires Reapproval>=1 or Total Number of Rejections>=1)) {
    Update Linked Approval Records to Pend Approval
}
```

Create: All Creation Actions

Purpose: This rule handles all creation actions, and emails the Document Manager if the initial Status is Pending Review.

Runs: When created by email, web or API.

Search Filter: None.

Priority: 1

Actions:

- ◼ I: All Creation Actions

```
if (Status='Pending Review') {
    Email Document Mgr that Document is Pending Review
}
```

Edit: Handle Rejection of an Approval (Web or API)

Purpose: This rule handles all actions if an Approval is rejected.

Runs: When edited by web or API.

Search Filter: R: Has one or more rejections and status is not draft

Priority: 4

Actions:

- ◼ U: Set Status to Draft
  - ◼ **Updated Field**: Status
  - ◼ **Updated Value**: Draft

- ■ E: Email Submitter Document Requires Update
  - ■ **Sent To:** Submitter
  - ■ **Template Name:** Your Document Requires and Update
- ■ L: Set Document Rejected field to Yes in linked Approvals
  - ■ **Chain:** Document > Approvals through **fields: Document ID, Document Title, e**tc.
- ■ U: Set Document Rejected to Yes
  - ■ **Updated Fields:** Document Rejected, Requires Update (Approval**s** table)
  - ■ **Updated Value**: Yes

Edit: Refresh Concurrent Approvals if Total Number of Approvals Changes (web, API)

Purpose: This rule handles updates to approval records when the Total Number of Approvals changes.

Runs: When created or edited by Web or API.

Search Filter: R: Total Number of Approvals Changed

Priority: 20

Actions:

- ■ U: Refresh Concurrent Approvals
  - ■ **Chain:** Document > Approvals through Document ID
  - ■ **Updated Field:** Concurrent Approvals
  - ■ **Updated Value:** The records that result from the search MT: Approvals for same contract or document with same step number

Action Buttons

Submit for Review	
Purpose	This button will change the status of the record to Pending Review and save the record.
Visibility	Only visible when Status is Draft or Retired.

Submit for Approval	
Purpose	This button will change the status of the record to Pending Approval following validation, set Last Document ID field in Employee record and save the record.

Visibility	Only visible when Status is Pending Review.

Cancel	
Purpose	This button will change the status of the record to Cancelled and save the record.
Visibility	Only visible when Status is Draft.

Publish	
Purpose	This button will change the status of the record to Published and save the record.
Visibility	Only visible when Status is Ready for Publication.

Publish without Approval	
Purpose	This button will change the status of the record to Published following validation and save the record.
Visibility	Only visible when Status is Pending Review.

Mark Retired	
Purpose	This button will change the status of the record to Retired and save the record.
Visibility	Only visible when Status is Published.

Return to Submitter	
Purpose	This button will change the status of the record to Draft, email the submitter to update the document and save the record.
Visibility	Only visible when Status is Pending Review.

Incidents Table Rules

Rules and actions:

Set Priority on New Tickets

Saved Search: R: Priority is Empty

- ▪ I: Set Priority

```
if (Impact='Affects Single User') {
    if (Urgency='Low') {
        Set Priority to Low
    } else if (Urgency='Medium') {
        Set Priority to Medium
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to High
    }
} else if (Impact='Affects Multiple Users' or Impact='Affects Department') {
    if (Urgency='Low') {
        Set Priority to Low
    } else if (Urgency='Medium') {
        Set Priority to Medium
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
} else if (Impact='Affects Office') {
    if (Urgency='Low') {
        Set Priority to Medium
    } else if (Urgency='Medium') {
        Set Priority to High
    } else if (Urgency='High') {
        Set Priority to High
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
} else if (Impact='Affects Company') {
    Set Priority to Critical
    if (Urgency='Low') {
        Set Priority to High
    } else if (Urgency='Medium') {
        Set Priority to High
    } else if (Urgency='High') {
        Set Priority to Critical
    } else if (Urgency='Critical') {
        Set Priority to Critical
    }
}
```

Incident - All Creation Actions

- ■ I: Conditional Creation Actions

```
if (Spawning SR ID!=NULL) {
      Update SR with conversion note
}
if (Problem CI Identified?='Yes' and Assigned Team=NULL and CI Responsible Team!=NULL) {
      Assign to CI Responsible team
} else {
      Assign to 1st level support team
}
```

- ■ I: Notify Customer and Assignee

```
if (Status!='Closed') {
      Send Submitter creation acknowledgment email
      if (Assigned Person!=NULL) {
            Email Assigned Person of new Incident
      } else {
            Email Assigned Team of new Incident
      }
}
```

Incident - Edit actions (API Enabled)

- I: Notification Actions for Incidents

```
if (Problem Closed? changed from: any value to: 'Yes' during record's last modification) {
      Email Incident Assignee of Problem Closing
}
if (Updated By!='$assigned_person' and Assigned Person changed from: any value to: any value during record's last
modification or Assigned Team changed from: any value to: any value during record's last modification) {
      if (Assigned Person!=NULL) {
            Email Assigned Person of new assignment
      } else {
            Email Assigned Team of new Incident
      }
      if (Assigned Team changed from: any value to: any value during record's last modification) {
            Increment Number of Assignees
      }
      if (Assigned Person changed from: any value to: any value during record's last modification) {
            Increment Number of Teams Assigned
      }
}
if (Status changed from: any value to: 'Workaround Provided' during record's last modification) {
      Email submitter and assignee of Workaround
}
```

- I: Conditional Assignment Actions

```
if (Assign to CI Responsible Team changed from: any value to: 'Yes' during record's last modification and CI Responsible
Team!=NULL) {
      Set Assigned Team to CI Responsible Team
}
```

Incident Customer Update Actions

Run upon Edit, no API

Saved Search: Updated by Submitter

- I: Submitter Update Actions

```
if (Status='Closed' and I would like to reopen my Incident='Yes') {
      Change Status from Closed to Reopened
      Increment # of Reopens field
} else if (Status='Pending Customer') {
      Change Status to Update by Customer
}
if (Assigned Person!=NULL) {
      Email Assigned Person submitter update
} else {
      Email Assigned Team submitter update
}
```

Action Buttons

Add Time	
Purpose	Triggers action to convert Time Spent and Time Description fields to a Time Entry record linked to the current Change Request record, then blanks out Time Spent and Time Description.
On click	Save record, then execute action.
Actions	▪ I: Add Time

```
if (Time Spent=NULL) {
        Time spent must not be blank
} else {
        Convert to Time Entry
        Set Time Spent and Time Description to Blank
}
```

Convert to Change Request	
Purpose	Saves changes to the Incident and maps some of its fields into a new Change Request.
On click	Save record, then execute action.

Actions	▪ C: Convert to Change Request
	    ▪ **Conversation:** Change Request
	    ▪ **Options:** Interactive, showing user the new record page

Convert to Problem	
Purpose	Save before and after conversion, prevent conversion if the problem ID already has a value.
On click	Save record, execute action, save record again.
Actions	▪ I: Convert to Problem if first conversion

```
if (Linked Problem ID!=NULL) {
        Only one linked Problem allowed
} else {
        Convert to Problem
}
```

Convert to Service Request	
Purpose	Silently convert to service request.
On click	Execute action, then save record.

Actions	▪ C: Convert to Service Request from Incident
	▪ **Conversion:** Service Request
	▪ **Options:** Silently with no confirmation

Set Priority

Purpose	Automatically sets priority based on impact and urgency.
On click	Execute action.
Actions	▪ I: Set Priority

```
if (Impact='Affects Single User') {
        if (Urgency='Low') {
                Set Priority to Low
        } else if (Urgency='Medium') {
                Set Priority to Medium
        } else if (Urgency='High') {
                Set Priority to High
        } else if (Urgency='Critical') {
                Set Priority to High
        }
} else if (Impact='Affects Multiple Users' or Impact='Affects Department') {
        if (Urgency='Low') {
                Set Priority to Low
        } else if (Urgency='Medium') {
                Set Priority to Medium
        } else if (Urgency='High') {
                Set Priority to High
        } else if (Urgency='Critical') {
                Set Priority to Critical
        }
} else if (Impact='Affects Office') {
        if (Urgency='Low') {
                Set Priority to Medium
        } else if (Urgency='Medium') {
                Set Priority to High
        } else if (Urgency='High') {
                Set Priority to High
        } else if (Urgency='Critical') {
                Set Priority to Critical
        }
} else if (Impact='Affects Company') {
        Set Priority to Critical
        if (Urgency='Low') {
                Set Priority to High
        } else if (Urgency='Medium') {
                Set Priority to High
        } else if (Urgency='High') {
                Set Priority to Critical
        } else if (Urgency='Critical') {
                Set Priority to Critical
        }
}
```

Insurance Certificates Table Rules

Rules and actions:

TB: Handle Insurance Certificate Expiration (Disabled by Default)

Purpose: Sets the status to Expired for contracts that were Active and past their expiration dates, and catches any other contracts past expiration and sets their status to Inactive.

Runs: Every 2 days at 2 AM

Saved Search: Valid Certificates Expiring Today

Actions:

- ■ I: Handle Certs Expiring Today

```
if (Contract Status='Active') {
        Set Status to Expired
        Email Contract Manager of expired certificate
} else {
        Set Status to Inactive
}
```

TB: Update Contracts (daily) (disabled)

Purpose: This rule updates the linked field of Contracts to find all contracts related to the insurance certificate.

Runs: Daily at 1:00 AM

Search Filter: R: Status is Valid

Priority: 0

Action:

- ■ U: Set Default Contracts
  - ■ **Updated Field:** Contracts
  - ■ **Updated Value:** Use the Insurance Certificate to Contract: Status is greater than or equal to Active and Company matches search, and choose all records**.**

Items Requested Table Rules

Action buttons:

Assign to User

Purpose: Update Configuration Item record with appropriate data.

On Click: Execute action, then save record and open for editing

Actions:

- L: Assign to User – linked update of CI
    - **Chain:** Item Requested/CI Name -> Configuration Item
    - **Linked Action:** U: Copy Req. Login to CI User Login, Update Status and Ownership
- U: Copy Req. Login to CI User Login, Update Status and Ownership
    - **Updated Fields:** Ownership, Status, User Login
    - **Updated Values:** Assigned to an Internal User, Pending Install, "$item_requested.requester_login"

Leads Table Rules

Rules and actions:

Lead Conversion

Purpose: Runs conversion of Lead to Company, Opportunity, and Contact records when Status of Lead record is set to "Converted."

Runs: Edit by Email, Web or API

Search Filter: Just Converted

Priority: 2

Actions:

- C: Convert to Company, Opportunity, Contact
    - **Conversions:** People, Opportunity, Company
    - **Options:** Silently with no confirmation

Campaign actions

Purpose: Automatically sets campaign name from name of lead.

Runs: Create by Email, Web, API

Search Filter: R: No campaign but user entry

Priority: 1

Actions:

- U: Update Campaign from How did you hear
    - **Field Changed:** Campaign Name
    - **Value Changed to:** $lead_source

Action Buttons

Find Local Hotels	
Purpose	Searches Google Maps for hotels near the lead.
On click	Opens URL: "http://maps.google.com/maps?f=q&q=hotels" + "loc:" + urlEncode($street_address_1) + "," + urlEncode($city) + "," + $us_state
Map It	

Purpose	Searches Google Maps to find the location of the lead.
On click	Open URL: "http://maps.google.com/maps?f=q&q=" + urlEncode($street_address_1) + "," + urlEncode($city) + "," + $us_state

People Table Rules

Rules and actions:

Edit: Approval Record Creation

Purpose: Performs conversion of Employee records to Approval records for documents approval and updates fields keeping track of Approvals that have been created.

Runs: Edit by Web or API

Search Filter: R: Last Doc ID Text Changed and All Docs does not contain Last Doc ID

Priority: 1

Actions:

- C: Convert Employee to New Approval
  - **Conversion:** Approval
  - **Options:** Silently with no confirmation
- U: Add to All Document IDs Converted
  - **Fields Changed:** All Document IDs Converted
  - **Value Changed to:** $last_document_id_text+","

Action Buttons

Export to Outlook	
Purpose	Downloads the user's information to vcf format which will allow it to be saved and opened in MS Outlook
On click	Open URL: "/gui2/vCard;jsessionid=?id=" + urlEncode($id)

Problems Table Rules

Rules and actions:

Rule: Problem - Creation actions

- I: Conditional Creation Actions

```
if (Spawning Incident ID!=NULL) {
      Update "Additional Info" in Spawning Incident
}
```

Rule: Problem - Edit actions (API Enabled)

- I: Notification actions

```
if (Status changed from: any value to: 'Resolved' during record's last modification) {
      if (Spawning Incident ID!=NULL) {
            Set "Problem Closed?" in Incident to Yes
      }
}
if (CR Closed? changed from: any value to: 'Yes' during record's last modification) {
      Email Problem Assignee of CR Closing
}
```

Action Buttons

Add Time	
Purpose	Triggers action to convert Time Spent and Time Description fields to a Time Entry record linked to the Service Request, then blanks the fields.
On click	Save record, then execute action.

Actions	I: Add Time
	``if (Time Spent=NULL) {
	      Time spent must not be blank
	}
	Convert to Time Entry
	Clear Time fields```

Convert to Change Request	

Purpose	Validates the problem has no change request associated with it, then converts to a change request.
On click	Execute action, then save record.

Actions	◾ I: Convert to CR (if first conversion)
	
	``
	if (Change Request ID!=NULL) {
	        Cannot have more than one linked CR
	} else {
	        Convert to Change Request
	}
	``

Copy Solution to Incident and Close It	
Purpose	Validates a resolution exists, then copies the resolution from the problem to a linked incident.
On click	Execute action.

Actions	◾ I: Copy Solution field to related Incidents
	
	``
	if (Resolution=NULL) {
	        solution must have content
	} else {
	        Copy Solution from Problem to Incident
	}
	``

Copy Workaround to Incident	
Purpose	Copies the contents of the Workaround field in the Problem record to the Resolution fields of linked Incidents. Sets the "Workaround Provided?" field in the Problem record to Yes.
On click	Execute actions.

Actions	◾ I: Update Linked Incident with Workaround
	
	``
	if (Workaround=NULL) {
	        you must have content in workaround field
	} else {
	        Copy Workaround from Problem to Incident
	}
	``
	
	◾ U: Set Workaround Provided to yes
	   ◾ **Updated Field:** Workaround Provided
	   ◾ **Updated Value:** Yes

End Diagnosis Clock	
Purpose	Sets Diagnosis Finish Time to "now."
On click	Execute action.

Actions	■ U: Set Diagnosis Finish Time to Now
	    ■ **Updated Field:** Diagnosis Finish Time
	    ■ **Updated Value:** NOW()

Set Solution Finish Time

Purpose	Sets Solution Finish Time to NOW().
On click	Execute action.

Actions	■ U: Set Solution Finish Time to Now
	    ■ **Updated Field:** Solution Finish Time
	    ■ **Updated Value:** NOW()

Set Solution Start Time

Purpose	Sets Solution Start Time to Now
On click	Set Solution Start Time to Now

Actions	■ U: Set Solution Start Time to Now
	    ■ **Updated Field:** Solution Start Time
	    ■ **Updated Value:** NOW()

Start Diagnosis Clock

Purpose	Sets Diagnosis Start Time to NOW().
On click	Execute action.

Actions	■ U: Set Diagnosis Start Time to Now
	    ■ **Updated Field:** Diagnosis Start Time
	    ■ **Updated Value:** NOW()

Projects Table Rules

Rules and actions:

Edit by Web: All edit actions: trigger tasks, cancel tasks when cancelled, etc.

Purpose: Validates, triggers, and cancels tasks.

Runs: On Edit by Web.

Search Filter: None.

Priority: 2

Actions:

- I: All Edit Validations

  ```
  if (Status changed from: any value to: any of 'Completed','Tasks Completed' last user's modification and Number of Open Tasks>=1) {
        cannot complete while open tasks
  }
  ```

- I: All Edit Actions

  ```
  if (Default Tasks!=NULL and Status changed from: any value to: 'Work In Progress' last user's modification) {
        Trigger Default Tasks
  } else if (Status changed from: any value to: 'Cancelled' last user's modification) {
        Cancel Tasks
        Email Project CCs about Cancelled Project
  }
  ```

TB: notify Project Manager if hours exceed authorized hours

Puprose: Checks on authorized hours and hours worked and sends notification if authorized hour are exceeded.

Runs: Every day.

Search Filter: R: total hours for client project exceeds authorized hours

Priority: 1

Actions:

- U: Set Status to Assigned
  - **Fields Updated:** Status
  - **Updated To:** Assigned
- E: Email PM project has exceeded hours
  - **Email Template:** Project $formula($project_name) has exceeded authorized hours

Edit by Web or API: Handle completion or addition of tasks and status changes

Purpose: Opens or completes projects depending on number and state of tasks linked to project.

Runs: Edits by Web or API.

Search Filter: R: Has Tasks and the number of open tasks changed

Priority: 1

Actions:

- I: Handle completion or adding tasks

```
if (Number of Open Tasks=0 and Status='Work In Progress') {
      Set Status To Tasks Completed
      Email PM, Project CCs All Tasks Completed
} else if (Number of Open Tasks changed from: =0 to: > 0 during record's last modification and Status='Tasks Completed')
{
      Set Status to Work in Progress
}
```

Create: All Project creation actions – trigger tasks

Purpose: Notify Project Manager and CCs of new project, and begin default tasks.

Runs: Creation by Email, Web, or API:

Search Filter: None.

Priority: 1

Actions:

- I: All project create actions

```
if (Creator Login!='$pm_login') {
      Email PM of new project assignment
}
if (Project CCs!=NULL) {
      Email project ccs of new project
}
if (Status='Work In Progress') {
      Trigger Default Tasks
}
```

Project Types Table Rules

Rules and actions:

Create/Edit by Web or API: Update Has Task Templates

Purpose: Checks to see if Task Templates exist in this Project Type and sets the flag field Has Task Templates to Yes if so.

Runs: Create/Edit including rules by Web or API

Search Filter: R: Has task templates is no and number of templates is 1 or more

Priority: 10

Actions:

- U: Set Has Task Templates to Yes
    - **Field Changed:** Has Task Templates
    - **Value Changed to:** Yes

Purchase Orders Table Rules

Action buttons:

Calculate Value

Purpose: Multiplies the Hours by the Rate, and puts the result in the field "Total Value".

On Click: Execute action.

Action:

- U: Total Value = Hours * Rate
    - **Updated Field:** Total Value
    - **Updated Value:** $hours * $rate

Purchase Requests Table Rules

Rules and actions:

Rule: Purchase Request - New Request Actions

Condition and Schedule: When a Purchase Request is Created by Web, Priority 1

- ■ I: Validation for New Requests

  ```
  if (Supervisor Approved='Yes' and Created By='$approved_rejected_by0') {
        Only Supervisor Can Approve
  }
  ```

- ■ I: New Purchase Request Actions

  ```
  Email Requestor about new Purchase Request
  Email Purchasing Team aboue new Purchase Request
  Email Supervisor that PR requires approval
  if (Supervisor Approved='Yes') {
        Set Status to "Supervisor Approved"
  }
  ```

Rule: Purchase Request - Edit Validation Actions

- ■ I: Actions Based on Edit Validations

  ```
  if (Status='Approved by Procurement' and (Supervisor Approved=NULL or Procurement Group Approved=NULL)) {
        Validate Approved Status Change
  }
  if (Status changed from: any value to: 'Rejected by Procurement' last user's modification and Additional Notes not changed
  last user's modification) {
        Validate Additional Notes exist for Not Approved request
  }
  ```

Rule: Purchase Request - All Staff Edits

- ■ I: Staff Edit Actions

```
if (All Items Received?='Yes' and Status!='Received') {
      Update Status to Received
}
if (Status!='Supervisor Rejected' and Supervisor Approved changed from: any value to: 'No' last user's modification) {
      Set Status to Supervisor Rejected
}
if (Status changed from: any value to: 'Rejected by Procurement' last user's modification) {
      Email Requestor about Not Approved Request
}
if (Supervisor Approved changed from: any value to: 'Yes' last user's modification and Status='Pending Approval') {
      Set Status to Supervisor Approved
}
```

Rule: Purchase Request - All End User Edits

- ■ I: Actions Based on End User Edits

```
if (Additional Notes changed from: any value to: any value last user's modification and Requester Login='$updater_login') {
      Email Purchasing Team about updated Purchase Request
}
```

Service Requests Table Rules

Rules and actions:

Rule: All new Service Request actions

Runs when created by any method, no saved search

- ▪ I: Assign Service Request

```
if (Give Service Team Priority over CI Team!='Yes' and Assigned Team=NULL and CI Responsible Team!=NULL) {
      Set assigned team to ci responsible team
} else if (Assigned Team=NULL and Service Responsible Team!=NULL) {
      Set Assigned team to service responsible team
} else {
      Set assigned team to 1st level support team
}
```

- ▪ I: Trigger Creation of tasks from template

```
if (New Employee Tasks!=NULL and Additional Fields to Show~='New Employee Tasks field') {
      Update Last SR ID field in linked templates
}
```

- ▪ I: Send emails to customer and assignee

```
if (Status!='Closed') {
      acknowledge to customer
      if (Assigned Person!=NULL) {
            email assigned person of new assignment
      } else {
            Email assigned team of new SR
      }
}
```

Rule: All edit validations and other actions

Runs when edited, no selected search

- ▪ I: All edit validations by staff

```
if (Status changed from: any value to: 'Pending Customer' last user's modification and Additional Information not changed
last user's modification) {
      Must put comments for pending customer
}
if (Status changed from: any value to: 'Closed' last user's modification and Resolution=NULL) {
      must have resolution
}
```

- I: Trigger new task creation

```
if (New Employee Tasks changed from: any value to: any value last user's modification) {
     Update Last SR ID field in linked templates
}
```

Rule: All Customer Update Actions

- I: User Update Actions

```
if (Status='Closed' and I would like to reopen my Service Request='Yes') {
     Change Closed to Reopened
     Update # of time reopened field
} else if (Status='Pending Customer') {
     Change to Updated by Customer
}
if (Assigned Person!=NULL) {
     Email assignee customer update
} else {
     Email Assigned Team of customer update
}
```

Rule: Tasks Just Completed

Runs on edit, Search: R: Completed Tasks just changed to All

- I: Notify Assigned team or person of completed tasks

```
if (Assigned Person!=NULL) {
     Email assigned person tasks are completed
} else {
     Email Assigned Team that tasks are completed
}
```

Rule: Status Change Actions

Runs on edit, Search: R: Status changed last user mod

- I: All Status Change Actions

```
if (Status changed from: any value to: 'Closed' last user's modification) {
    Reset reopen flag
    if (Priority='Critical') {
        if (Total Hours to Close>'$sla_critical_priority_hours_to_complete') {
            Set "Solved within SLA" to No
        }
    } else {
        if (Total Hours to Close>'$sla_hours_to_complete') {
            Set "Solved within SLA" to No
        }
    }
}
```

Rule: Assigned team or person changed

Runs on edit, Search: Assigned person or team changed

- I: Notify team or person of new assignment

```
if (Assigned Person!=NULL) {
    email assigned person of new assignment
} else {
    Email assigned team of reassignment
}
if (Assigned Team changed from: any value to: any value last user's modification) {
    Increment number of teams assigned
}
if (Assigned Person changed from: any value to: any value last user's modification) {
    Increment number of assignees
}
```

Action Buttons

Add Time	
Purpose	Triggers conversion mapping Time Spent and Time Description fields onto a new Time Entry record linked to this Task then blanks those fields.
On click	Execute action.
Actions	- I: Validate and Create Time Entry

```
if (Time Spent=NULL) {
    Time spent must not be blank
} else {
    Convert to Time Entry
    Blank out time fields in SR
}
``` |

Clone	
Purpose	Makes a copy of the Service Request, currently unused.
On click	Execute action.
Actions	- C: Clone to Service Request **Conversion:** Service Request

Convert to Change Request	
Purpose	Convert to create a new Change Request, then save and close record.
On click	Execute action then save record.
Actions	I: Convert to Change Request If First One

```
if (CR ID!=NULL) {
        can only convert to cr once
} else {
        Convert to Change Request
}
```

Convert to Incident	
Purpose	Validates that Problem has not been converted to Incident before, then saves and converts to an Incident.
On click	Execute action then save record.
Actions	I: Convert to Incident If First Conversion

```
if (Incident ID!=NULL) {
        Can only convert to Incident once
} else {
        Convert to Incident
}
```

Escalate	
Purpose	Sets Assigned Team to 2nd Level Support Team and Priority to Critical.
On click	Execute action.
Actions	- U: Set Assigned Team to 2nd Level Support Team and Priority to Critical **Updated Fields:** Assigned Team, Priority **Updated Values:** 2nd Level Support Team, Critical

Generate Tasks	
Purpose	Generates the tasks based on the service details.
On click	Executes action.

Actions	■ I: Create Task Actions
| | ```
if (Task Generation Method='Predefined Task Workflow') {
    if (Task Workflow ID=NULL) {
        There is no workflow defined
    } else {
        Trigger Required and Conditional Tasks and set flag
        Set Show Task Generation Button to No
    }
} else if (Task Generation Method='Ad Hoc or Single Task') {
    if (Task to Generate ID=NULL) {
        There is no template defined.
    } else {
        Update linked Task Template with SR ID and set flag
    }
} else if (Task Generation Method='User Selected Tasks' and Select New Tasks!=NULL) {
    L: Update Latest SR ID field in linked templates and set flag
}
if (Status='Open') {
    Set Show Task Launch Button to Yes
}
``` |

Support Cases Table Rules

Rules and actions:

Assignee Change by someone else

Purpose: Informs assignee when assignee is changed

Runs: Edit by Email, Web, or API

Search Filter: Assignee change

Priority: 1

Actions:

- E: Email assignee
  - **Template:** Case #$id assigned to you: "$summary"

SC – All new case actions

Purpose: Inform assignee or assigned team of new case.

Runs: Create by Email, Web, or API

Search Filter: None

Priority: 2

Actions:

- I: Email Assigned team or person of new case

```
if (Assigned Person!=NULL) {
        email assignee new case
} else if (Assigned Team!=NULL) {
        Email Assigned team new case
}
```

SC – All Customer Update Actions

Purpose: Handles status changes and email notices from interactions with customer.

Runs: Edit by Email or Web

Search Filter: R: Updater Team is customer team

Priority: 3

Actions:

- I: Customer Update Ticket Actions

```
if (I Would Like To Reopen My Ticket changed from: any value to: 'Yes' last user's modification and Status='Closed') {
      Change Status To Reopened
}
if (Status='Sent to Customer') {
      Set State to Updated by Customer
}
if (Assigned Person!=NULL) {
      email assignee update
} else {
      Email Assigned Team of Customer Update
}
```

Action Buttons

Convert to Change Request	
Purpose	Check to see if Change Request already exists for this Support Case. If not, converts to create a Change Request from Support Case.
On click	Do nothing, execute actions, then save and open record for view.

Action	- I: Convert to change request if not yet done
	
	 ```if (CR ID!=NULL) {```
	 ``` can only convert sw once```
	 ```} else {```
	 ``` Convert to Change Request```
	 ```}```

Tasks Table Rules

Rules and actions:

Create: All create validations

Purpose: Restricts addition of tasks to legal statuses and warns the user if the due date is earlier than the time it is created (the user may pass through this warning). Note that this will not run on tasks created by API.

Runs: Create by Web.

Search Filter: None

Priority: 1

Actions:

- I: All create validations

```
if (Project Status='Completed' or Project Status='Cancelled') {
      Cannot add task to completed or cancelled project
}
if (Date Due<this day) {
      warn that due date is in the past
}
```

Create: All Creation Actions

Purpose: Sets due dates, assigns project manager and teams, and emails assignees of assignation.

Runs: Create by Email, Web, API:

Search Filter: None
Priority: 1

Actions:

-

Selected Actions

I: All Creation Actions
I: Set Assigned Team
I: Email Assignees About New Assigned Task
I: Set Sequence to 1 for single tasks
I: Set Prerequisite tasks for those from template
U: Set Highest Sequence

- I: All Creation Actions

This action has several parts based on the type of request the task is associated with. In general, it does not permit a comma in the title and strips it out because the comma breaks the prerequisite task handling. It also sets the Status to Conditional for any task whose Task template usage is Conditional. The other actions handle setting the assignment and due dates based on the context:

```
if (Task Template Usage='Conditional') {
    Set Status to Conditional
}
if (Related To='Service Request') {
    if (Template Assign Based On='Person from Service Request') {
            Set Assigned Person from Variable
    } else if
    (Template Assign
    Based On='Team
    from Service
    Request') {
            Set Assigned Team from Variable
    }
    if (Template Number of Hours to Due Date>0 and Date Due=NULL and Status='Assigned' and Source Template ID!=NULL)
            Set Date Due to Date Updated plus Number of Working Hours to Due Date
    } else if (Date
    Due=NULL and
    Status='Assigned')
    {
            Set Date Due to Now + 2 Days
    }
    if (SR Status<='In Progress') {
            Set Launched to Yes
            if (Number of Prerequisite Tasks=Number of Completed Prerequisite Tasks or Number of Prerequisite Tasks=0) {
                    if (Task Template Usage!='Conditional' or Template Condition~='true') {
                        Set Status to Assigned
                    } else {
                        Set Status to Not Needed
                    }
            }
    }
}
```

```
if (Related To='Projects') {
    if (Template Assign Based On='Person from Project') {
            Set Assigned Person from Variable
    } else if
    (Template Assign
    Based On='Team
    from Project') {
            Set Assigned Team from Variable
    }
    if (Template Number of Hours to Due Date>0 and Date Due=NULL and Status='Assigned' and Source Template ID!=NULL) {
            Set Date Due to Date Updated plus Number of Working Hours to Due Date
            // checking that it started from a template prevents overwriting of manually entered due date entered when ad hoc t
    }
    if (Project Status<='Work In Progress') {
            Set Launched to Yes
            if (Number of Prerequisite Tasks=Number of Completed Prerequisite Tasks or Number of Prerequisite Tasks=0) {
                    if (Task Template Usage!='Conditional' or Template Condition~='true') {
                        Set Status to Assigned
                    } else {
                        Set Status to Not Needed
                    }
            }
    }
    Refresh task calculation fields in project
}
```

```
if (Related To='Change Request') {
    if (Template Assign Based On='Team from Change Request') {
            Set Assigned Team from Variable
    } else if
    (Template Assign
    Based On='Person
    from Change
    Request') {
            Set Assigned Person from Variable
    }
    Set Due Date to Change Window End Time
    if (Change Status<='In Progress') {
            Set Launched to Yes
            if (Number of Prerequisite Tasks=Number of Completed Prerequisite Tasks or Number of Prerequisite Tasks=0) {
                    if (Task Template Usage!='Conditional' or Template Condition~='true') {
                        Set Status to Assigned
                    } else {
                        Set Status to Not Needed
                    }
            }
    }
}
if (Task Summary~=',') {
    Strip comma out of task summary
}
```

- I: Set Assigned Team

  Purpose: This sets the assigned team to the primary team of the assigned person if the task has been assigned to a specific person from the request. If for some reason the system has tried to assign to a specific person and that person is not defined so there is no assigned team or person, then the system sets the Assigned Team to the 1st Level Support Team:

```
if (Assigned Team=NULL and Assigned Person!=NULL) {
    Set Assigned Team to Assigned Person Primary Team
} else if (Assigned Team=NULL and Assigned Person=NULL) {
    U: Set the Assigned Team to 1st Level Support Team
}
```

- I: Email Assignees About New Task

  The assignee is only emailed if the task is already launched and set to a Status of Assigned. Otherwise the notification is sent when the status does change:

```
if (Assigned Person!=Created By) {
    if (Assigned Person!=NULL and Status='Assigned') {
        Email Assigned Person New Task
    } else if (Assigned Team!=NULL and Status='Assigned') {
        Email Assigned Team New Task
    }
}
```

- I: Set Sequence to 1 for single tasks

  Purpose: For change request tasks that may be drawn from a task template that is part of a task workflow and that might therefore have a higher sequence value, this rule sets the Sequence value to 1:

  ```
  if (Related To='Change Request') {
      if (Change Service Task Method='CR Single Task from Template' and Source Template ID!=NULL) {
          Reset Step Number to 1
      }
  }
  ```

- I: Set Prerequisite tasks for those from template

  Purpose: If this is a task that is based on a task template that might have prerequisite task templates, it runs an action to set the Prerequisite tasks based on those whose source template IDs match the Prerequisite IDs in this task's source template.

  ```
  if (Source Template ID!=NULL) {
      Set Prerequisite Tasks
  }
  ```

- U: Set Highest Sequence

  Purpose: This action sets the linked field to the Prerequisite Task to this task that has the highest sequence value. This is used to set this task's sequence in the next rule to 1 higher than that value.

Create/Edit: Update Sequence based on Highest Sequence

Purpose: This sets the Sequence value for a new task to the next highest number after all of its prerequisite tasks to indicate where in the general sequence it will be due.

Runs: Create/Edit (including edits made by other rules during creation) by Web, API:

Search Filter: R: Highest Sequence is not empty and changed last modification

Priority: 2

Actions:

- U: Set Sequence to Highest Prereq plus 1

Edit: All Edit Actions (API enabled)

Purpose: Handles updating a task when it is launched to a status of Assigned – sets the due date and notifies the assignees; When a task is completed, handles the launching of any tasks for which it was the prerequisite, sets the alert color, and notifies the appropriate people, and also handles pushing any working notes into the running notes fields in the related record.

Runs: Edit by Web or API

Search Filter: None

Priority: 4

Actions:

- I: All Edit Actions (API enabled)

```
if (Status changed from: any value to: 'Assigned' during record's last modification) {
    if (Source Template ID!=NULL and Template Number of Hours to Due Date>0 and Date Due=NULL) {
        Set Date Due to Date Updated plus Number of Working Hours to Due Date
    } else if
    (Date
    Due=NULL)
    {
        Set Date Due to Now + 2 Days
    }
    if (Auto-Complete?='Yes') {
        Set Status to Done
        if (Notify for Auto-Completion?='Yes') {
            Email Assigned Team auto-completed notification
        }
    } else {
        if (Assigned Person!=NULL) {
            Email Assigned Person New Task
        } else {
            Email Assigned Team New Task
        }
    }
} else if (Status changed from: any value to: any of 'Completed','Failed','Not Needed' during record's last
modification) {
    if (Number of Dependent Tasks>=1) {
        Recalculate the Prerequisite Task Counts in Dependent Tasks
        // If prereqs are now done, separate rule triggers the dependent task
    }
    if (Assigned Person=NULL) {
        Set Assigned Person to person who closed
    }
    if (Alert Color='Red') {
        Set Alert Color to Default
    }
    if (Project Manager!=NULL and Project Manager!=Updated By) {
        Email PM when task is closed or canceled
    }
}
if (Working Notes changed from: any value to: any value last user's modification and Working Notes!=NULL) {
    Append Working Notes to Running Notes
    if (Related To='Change Request' and Change ID!=NULL) {
        Append Task Notes to Change Request
    }
    Blank Working Notes
}
```

Edit: Assign tasks when number of completed prerequisites meets criteria (API)

Purpose: This updates dependent tasks when their prerequisites are completed, checking if they are conditional, whether the condition is currently met.

Runs: Edit by Web, API:

Search Filter: R: Prerequisite tasks just completed and status is queued or conditional

Priority: 1

Actions:

- I: Assign or Mark as Not Needed based on condition

```
if (Task Template Usage!='Conditional' or Task Template Usage='Conditional' and Template Condition~='true') {
    Set Status to Assigned
} else if (Task Template Usage='Conditional') {
    Set Status to Not Needed
}
```

Edit: If task title has a comma, strip it out.

Purpose: Strips any commas that have been added to the task title. This is necessary because the task title needs to be compared against the prerequisite task titles and if they contain a comma, it breaks the comparison.

Runs: on Edit (including edits made by other rules during record creation) by Web or API

Search Filter: R: Task Summary contains comma

Priority: 40

Actions:

- U: Strip comma out of task summary

TB: (Disabled) Notify of upcoming task

Purpose: Notifies appropriate parties of pending tasks.

Runs: Daily

Search Filter: R: Upcoming Tasks

Priority: 3

Actions:

- I: Notify assigned person or team of due date

```
if (Assigned Person!=NULL) {
    Notify Assignee of Pending Activity
} else {
    Email Assigned Team of Pending Due Date
}
```

TB: (Disabled) Set alert color field

Purpose: Updates helper Alert Color field to red when criteria met.

Runs: Daily

Search Filter: R: Status not done, cancelled, or failed and due date is passed and not red

Priority: 30

Actions:

- U: Set Alert Color to Red
  - **Fields Changed:** Alert Color
  - **New Value:** Red

Action Buttons

Add Time	
Purpose	Validates that Time Spent has a value, then converts Time Spent and Time Description to a Time Entry and blanks out those fields.
On click	Save record, then execute action.
Visibility	If Enable Prerequisite Tasks is Yes.
Actions	■ I: Add Time

```
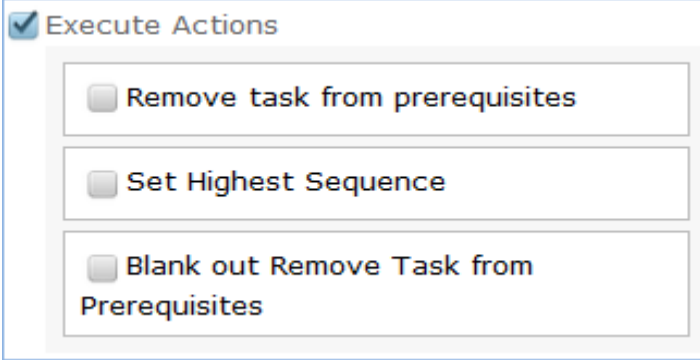if (Time Spent='NULL') {
    Time spent must not be blank
} else {
    Convert to Time Entry
    Set Time Spent and Time Description to Blank
}
```

Add to Prerequisites	
Purpose	Adds the selected task from the same request as a Prerequisite task, then sets the highest sequence in the current record, then blanks out the Prerequisite task to add field.
On click	Do nothing, then execute action, then do nothing. The new prerequisite task shows up below in the table of prerequisite tasks.

Actions	☑ Execute Actions
	  ☐ Append task to Prerequisite Tasks
	  ☐ Set Highest Sequence
	  ☐ Blank out Prerequisite task to add

End Clock	
Purpose	Sets the Work actual end time to Now.
On click	Execute action, then save and reopen for editing
Actions	U: Set Work Actual End Time to Now

Remove from Prerequisites	
Purpose	Removes the selected task from the prerequisites and recalculates the highest sequence value, then blanks out the selected task.
On click	Execute action, then does nothing
Visibility	If Enable Prerequisite Tasks is Yes.
Actions	

Set CI to in Planned Downtime	
Purpose	Sets the Operational Status of the linked CI to Planned Downtime
On click	Saves, then executes action, then saves and reopens for edit.
Visibility	Task Type=CI Change and Status=Waiting for Others, Assigned
Actions	▪ U: Set CI to In Planned Downtime

Set CI to Working	
Purpose	Sets the Operational Status of the linked CI to Working
On click	Execute action, then save and reopen for editing.
Visibility	Task Type=CI Change and Status=Waiting for Others, Assigned
Actions	▪ U: Set CI to Working

Start Clock	
Purpose	Sets the Work Actual Start Time to Now
On click	Execute action, then save and reopen for edit.
Actions	▪ U: Set Work Actual Start Time to Now

ButtonAB: Reassign Task	
Purpose	This is a button to put on the action bar associated with the Inactive Tasks related tables to make it easy to move a Task back into the active Tasks related table by setting its Status to Assigned.

On click	Execute action then save
Actions	▪ U: Set Status to Assigned

ButtonAB: Remove	
Purpose	Used to remove a task from use in a particular record (service request, project, or change request). The task is set to Not Needed and moved into the Inactive Tasks related table. It is still kept linked to the request in case it needs to be resuscitated.
On click	Execute action then save
Actions	▪ U: Set Status to Not Needed

Set Date Done	
Purpose	Sets the Date Done field to current date/time.
On click	Execute action.
Visibility	Status is Done.
Actions	▪ U: Set Date Done to Now ▪ **Updated Field:** Date Done ▪ **Updated Value:** NOW()

Task Templates Table Rules

Rules and actions:

Edit: Generate Tasks when triggered from Service Requests

Purpose: Generate tasks and perform upkeep on All SRs converted field.
Runs: Edit by Web or API
Search Filter: R: Latest SR ID changed and not yet converted
Priority: 2
Actions:

- I: Create task for Service Request

```
Set Flag to Create Task for SR to No
if (All SR IDs converted!~==Latest SR ID and All SR IDs converted=NULL) {
        if (Task Usage='Required' or Task Usage='Conditional' and Condition='TRUE') {
                Convert to Task for Service Request
                Copy SR ID into All SRs converted
                Blank Latest SR ID
        }
}
```

- C: Convert to Task
    - **Conversion:** Task
    - **Options:** Silently with no confirmation.
- U: Copy SR ID text into All SRs converted
    - **Field Changed:** All SR IDs converted
    - **Value Changed to:** $last_sr_text +","

Edit: Generate new tasks when triggered from Project

Purpose: Generate tasks and perform upkeep on converted projects.
Runs: Edit by Web or API
Search Filter: Project ID has changed and not yet converted
Priority: 1
Actions:

- C: Convert to Task
    - **Conversion:** Task
    - **Options:** Silently with no confirmation.

- U: Copy Project ID text into all projects converted
  - **Field Changed:** All Project IDs Converted
  - **Value Changed to:** $last_project_id_text +","

Edit: Generate Tasks when triggered from Change Request

Purpose: Generates tasks for a change request if they are required or conditional and the condition is true.

Runs: Edit by Web or API

Search Filter: LF: Flag to Create Task for CR from Template is Yes

- I: Create Task for Change Request

```
Set Flag to Create Task for CR to No
if (Task Usage='Required' or Task Usage='Conditional' and Condition='TRUE') {
      Convert to Task for Change Request
}
Blank Latest CR ID
```