

IBM Cúram Social Program Management



Customizing Plan Items

Version 6.05

IBM Cúram Social Program Management



Customizing Plan Items

Version 6.05

Note

Before using this information and the product it supports, read the information in "Notices" on page 19

Revised: May 2013

This edition applies to IBM Cúram Social Program Management v6.0 5 and to all subsequent releases unless otherwise indicated in new editions.

Licensed Materials - Property of IBM.

© **Copyright IBM Corporation 2012, 2013.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

© Cúram Software Limited. 2011. All rights reserved.

Contents

Figures	v
Tables	vii
Chapter 1. Introduction	1
1.1 Purpose	1
1.2 Audience	1
1.3 Prerequisites	1
Chapter 2. Why Customize a Plan Item? 3	
2.1 What is a Plan Item?	3
2.2 Why Customize Plan Items?	3
2.3 Creating a New Plan Item.	3
Chapter 3. Configuring a Plan Item 5	
3.1 Introduction	5
3.2 What Can You Configure?.	5
3.3 Integration with Cúram Workflow	7
3.4 Service Plan & Product Delivery Integration	7
3.5 Configuring Plan Item Cost	8
3.5.1 Estimated Cost	8
3.5.2 Actual Cost	9
3.6 Service Plan and Provider Integration	9
3.7 Integration with the Cúram Provider Management module	9
3.7.1 Update Client Participation:	11
3.8 New System Properties	11
Chapter 4. Implementing a Plan Item 13	
4.1 Introduction	13
4.2 Mileage Reimbursement Example.	13
4.2.1 Adding a Mileage Reimbursement Plan Item	13
4.2.2 Modifying the Mileage Reimbursement Plan Item	15
4.2.3 Viewing Mileage Reimbursement Plan Item Details	15
Chapter 5. Compliancy 17	
5.1 Miscellaneous Entities.	17
Notices 19	
Trademarks	21

Figures

Tables

1. Plan Item Configuration Items	5	5. System properties for 'service' and 'custom	
2. Plan Item - Product Delivery Relationships	7	service' type plan items.	11
3. CPM APIs	10	6. Product Delivery Creation Data.	14
4. Events raised by CPM	10		

Chapter 1. Introduction

1.1 Purpose

The purpose of this document is to explain to the reader how to design, implement and configure Service Plan Items.

In order to best understand these concepts, the guide should be read in full. The guide is not intended to be used as a training or user guide.

1.2 Audience

This document is intended for designers and developers who need to know how to extend Cúram Service Planning. Cúram Service Planning to include plan items that meet their specific requirements. A reasonable degree of knowledge of the application is assumed, however the reader must be very familiar with the concepts of Cúram Service Planning. For the most part, this document assumes a reasonable degree of technical knowledge from its readers; however, there are certain aspects of the document that may prove useful to business analysts that does not require technical knowledge.

1.3 Prerequisites

It is assumed that the reader is familiar with the basic concepts of Social Enterprise Management. In particular, it is assumed that the reader is familiar with the Cúram Case Manager and Cúram Service Planning.

Chapter 2. Why Customize a Plan Item?

2.1 What is a Plan Item?

Plan items are the fundamental building blocks of a Service Plan. They are the benefits, services and activities that are added to a plan in order to achieve the plan goal.

2.2 Why Customize Plan Items?

Cúram Service Planning can be applied to any of the industry segments within the SEM model. Social Enterprise agencies within these segments provide a diverse array of benefits and services to their clients from insurance/contribution based products to child support and medical assistance. As a result of this diversity the goals of service plans delivered by these agencies, and therefore the plan items used to achieve these goals, also vary greatly. To accommodate this Cúram Service Planning includes a framework that allows a customer to develop plan items to meet their specific requirements and to incorporate these seamlessly into a Service Plan.

2.3 Creating a New Plan Item

There are two aspects to this; the integration of a plan item with it's associated functionality (e.g. the creation of a specific type of case, the enactment of a particular workflow, the sending of a letter of referral etc.) and the configuration of the plan item to work with a Service Plan. The next chapter in this document outlines the configuration options that are available in Cúram Service Planning. The following chapter describes the process that must be followed to create a new type of plan item.

Chapter 3. Configuring a Plan Item

3.1 Introduction

The Cúram Service Planning Administration component allows the user to define and configure plan items.

3.2 What Can You Configure?

The following table lists the plan item attributes that are important from a configuration perspective.

Table 1. Plan Item Configuration Items

Attribute	Description
Reference	This is a user entered reference number for the plan item. This is required in order to identify the plan item if it must be referenced from any other business processing, e.g. from a rule set to impose sanctions if the plan item has not been completed successfully.
Type	The plan item type, e.g. Mileage Reimbursement plan item, Child Care Plan item etc. This is used to identify the type of plan item. Cúram Service Planning allows the user to link a plan item with a process/set of processes that should be undertaken in response to an event associated with the plan item or service plan. For example, a sample work flow is provided, which is enacted when a plan item of a type that has been associated to a Product Delivery is approved. This work flow contains a set of steps which, when executed, result in the creation of a product delivery case. The value of the type field is used as a transition condition in the service plan work flow to determine the path that the work flow should follow. If a customer implements custom code to undertake functionality defined for a plan item that they have created and the functionality must be invoked in response to a service plan event (e.g. plan item approval), then the required activities (or work flow) must be added to the appropriate service plan work flow.
Associated With	This optional field defines the type of object (product delivery, assessment, integrated case etc) that a service plan is associated with and is used to support integration between that object and the plan item. The first release of Cúram Service Planning includes integration between a plan item and a product delivery. Therefore, if the plan item results in the creation of a product delivery, in order to allow the plan item (and the service plan) and the product delivery to share information the value of this field must be set to Product Delivery.
Approval Required	This field is used to define whether the plan item requires approval. If this flag is set, a plan item of this type will be created with an initial status of Unapproved. However, if this flag is not set, then the plan item will be created with an initial status of Not Started.
Create Page Name	The name of the UIM page that is used to create a plan item of the type specified. As stated above, to meet the diverse requirements of plan items the customer must be able to extend Cúram Service Planning to add their own types of plan item. To support this, the customer must enter the name of the page used to create the plan item. The Cúram Service Planning framework includes pages that are used to identify the type of plan item that a user would like to add to a service plan. Once a user has selected a type of plan item, a resolve script is invoked that retrieves this create page name value for the plan item and opens the associated Java™ Server Page (JSP).

Table 1. Plan Item Configuration Items (continued)

Attribute	Description
Create Page Plan Item ID Parameter Name	This is the name of the parameter declared for the UIM page specified by the Create Page Name field (see above). This parameter identifies the service planning administration PlanItem record that defines the type of plan item to be added to the service plan. The value for this parameter is passed from the Create Plan Item Resolve Script to the UIM page that is used to create a new plan item.
Create Page Sub Goal ID Parameter Name	This is the name of the parameter declared for the UIM page specified by the Create Page Name field (see above). This parameter is used to identify the sub-goal to which the plan item is to be added. The value for this parameter is passed from the Create Plan Item Resolve Script to the UIM page that is used to create a new plan item.
Modify Page Name	The name of the UIM page that is used to modify a plan item of the type specified. On selection of a plan item for modification by a user, the service plan framework invokes the Modify Plan Item Resolve Script. This script uses the unique identifier for the plan item (this is the plannedItemID attribute of the PlannedItem record that is created when an instance of a plan item type is added to a service plan and should not be confused with the planItemID attribute of the PlanItem record that is used to define the plan item type) to determine the value of this field and then redirects the user to the specified page.
Modify Page Plan Item Parameter Name	This is the name of the parameter declared for the UIM page specified by the Modify Page Name field (see above) that must be set to the unique identifier for the plan item (this is the value of the plannedItemID attribute of the appropriate PlannedItem record). The value for this parameter is passed from the Modify Plan Item Resolve Script to the UIM page that is used to modify the plan item.
View Page Name	The name of the UIM page that is used to view the details of a plan item. On selection of a plan item for viewing by a user, the service plan framework invokes the View Plan Item Resolve Script. This script uses the unique identifier for the plan item (this is the plannedItemID attribute of the PlannedItem record that is created when an instance of a plan item type is added to a service plan and should not be confused with the planItemID attribute of the PlanItem record that is used to define the plan item type) to determine the value of this field and then redirects the user to the specified page.
Modify Page Plan Item Parameter Name	This is the name of the parameter declared for the UIM page specified by the View Page Name field (see above) that must be set to the unique identifier for the plan item (this is the value of the plannedItemID attribute of the appropriate PlannedItem record). The value for this parameter is passed from the Modify Plan Item Resolve Script to the UIM page that is used to modify the plan item.

From the above table it can be seen that a customer can define their own pages and functionality to create and maintain plan items. This allows Cúram Service Planning to support the diverse nature of plan items. However, plan items are typically about more than just a placeholder on a plan; there is usually an activity or task that must be undertaken. This is analogous to tasks on a standard project plan; the task on the plan describes the work that has to be done and how long it will take, however it is not the work itself. This is a separate activity that is undertaken outside of the plan. The result of this activity may be recorded on the plan, but the work is not done as part of the plan. This is where Cúram Service Planning differs from standard task orientated project planning. The work associated with the plan item may be completely or in some part undertaken by the system. Therefore there is not the same degree of separation between a plan item and the work to which it refers as there is for a task on a standard project plan. Cúram Service Planning includes a number of features to support the association between a plan item and the activities that must be undertaken in respect of it.

3.3 Integration with Cúram Workflow

As stated above the customer is allowed to define their own pages for the creation and maintenance of a plan item. Therefore the customer is also able to define the server processing that is invoked as a result of these pages. However there may be other points in the life cycle of a plan item or service plan that the customer would like to undertake functionality for a plan item, e.g. service plan approval, plan item approval or contract issue. To help achieve this, workflow events are raised from a number of places throughout Cúram Service Planning in response to specific service plan or plan item activities. Using Cúram Workflow the customer can map these events to workflow processes in order to undertake the required processing. For example, the addition of a plan item to a service plan that requires a product delivery case to supply the necessary benefits to the client should not actually create the case until the plan item has been approved. Therefore the create page defined for the plan item should result in the storage of data that makes it possible to create the product delivery upon plan item approval, however it should not create the case at this time. The product delivery case should be created when the plan item is approved. To achieve this, Cúram Service Planning raises a workflow event of type plan item approval from the 'Approve Plan Item' business process. Cúram Service Planning includes a workflow process definition that can be used for this purpose. This workflow process consists of a start activity and an automatic activity that is used to determine the value of the Type attribute of the associated plan item (the unique identifier for the plan item is passed as a parameter to the workflow as part of the raising of the workflow event). The value of the Type attribute can then be used to determine the path that the workflow should take (it is used as a condition in a workflow transition). The customer can then extend this workflow to include activities that should be undertaken for a specific type of plan item, or these activities can be packaged as a new workflow process that is enacted from the Cúram Service Planning workflow as a sub-flow.

3.4 Service Plan & Product Delivery Integration

The addition of a plan item to a service plan can in some instances result in the creation of a product delivery. Cúram Service Planning includes a number of interfaces designed to support integration between a plan item and a product delivery. There are several attributes of a plan item that have either a corresponding product delivery attribute or are related in some manner to particular aspects of product delivery functionality. The following table outlines these relationships.

Table 2. Plan Item - Product Delivery Relationships

Plan Item Attribute	Comment
Expected Start Date	The plan item expected start date is used to set the value of the case header startDate and expectedStartDate attributes. The expected start date of a plan item can not be set once the plan item has moved into a status of In Progress. The status of a plan item that is associated with a product delivery is set to In Progress when the product delivery is activated. Any change to the start date of the case prior to case activation will update the plan item expected start date to the same value. However, after the case has been activated, an update to the case start date will update the plan item actual start date and not the expected start date. Any update to the plan item expected start date prior to case activation will update the case header expectedStartdate and startDate attributes.
Expected End Date	The plan item expected end date is used to set the value of the case header expectedEndDate attribute. Any change to the expected end date of the case, prior to case activation, will update the plan item end date and vice versa. However, on case activation the expected end date of both the case and the plan item may not be modified.

Table 2. Plan Item - Product Delivery Relationships (continued)

Plan Item Attribute	Comment
Actual Start Date	The actual start date of the plan item is set to the start date of the product delivery at the time the product delivery is activated (setting the actual start date of the plan item will change its status to In Progress). The actual start date of the plan item should not be set until the case has been activated and this must be the only means by which this value can be set. The user should not be able to directly enter or update the value for the plan item actual start date. If the case start date changes after the case has been activated, then the plan item actual start date must also be updated.
Actual End Date	The actual end date of the plan item is set to the value of the case header endDate attribute when the case is closed. The user should not be able to directly enter or update the value for the plan item actual end date; it may only be set as a result of the case being closed. If the case is reactivated the plan end date is cleared.
Actual Cost	The actual cost for a plan item associated with a product delivery is derived from the payments made via the product delivery in respect of the benefit or service given to the client. The value of the actualCost attribute is cleared when a case is reactivated. The actual cost should not be entered by the user. Note, whilst the estimated cost is displayed during product delivery creation, it is not stored on the product delivery case and is used solely to help the case worker select the appropriate product provision.
Expected Outcome	The plan item expected outcome is used to set the value of the case header expectedOutcome attribute. Any change to the expected outcome of the plan item will update the product deliveries expected outcome until the time at which the product delivery is activated. From this point onwards the expected outcome may not be modified.
Outcome	The plan item outcome may only be entered upon completion of the plan item. A plan item may only be completed when an actual end date has been recorded. Therefore, in the case of a plan item with an associated product delivery, the product delivery must be closed before the outcome may be set. After the product delivery has been closed, the outcome may be set on the product delivery which will update the plan item outcome or vice versa.

3.5 Configuring Plan Item Cost

The estimated and actual cost can be maintained for a plan item. There are several different ways to derive the cost of a plan item, allowing for a variety types of plan items within a service plan.

3.5.1 Estimated Cost

The estimated cost is the cost an organization expects to pay for the delivery of a service associated with a specific type of service plan item. Depending on the business requirements for a plan item of a particular type, the estimated cost may be derived from the plan item definition, using the cost attribute of the PlanItem entity, or the cost can be entered by the user. For example, an organization has defined a Car Repair and a Bus Pass Provision plan item. The Car Repair plan item is intended to pay for repairs to a client's car to enable the client to travel to and from a place of work. The Bus Pass Provision plan item is intended to provide the client with a bus pass for one year, thus enabling the client to attend work. The bus pass provided by the organization has a fixed cost, whereas the cost of car repairs can vary considerably. Therefore, the plan item definition for the Bus Pass uses the PlanItem entity's cost attribute to define the estimated cost of the plan item to the organization (in fact, as this cost is fixed, this also defines the actual cost to the organization). The value of the PlannedItem entity estimatedCost and actualCost attributes are set to the value of the PlanItem entity cost attribute.

The pages defined to add and modify a plan item of this type should not allow the user to modify either the estimated cost or the actual cost. However, as any type of vehicle repair can be facilitated by the Car

Repair plan item, it is not possible for the organization to define a realistic estimated cost. Therefore the estimated cost to carry out the car repair must be entered by the user when the plan item is added to a service plan. The pages defined to add and modify a Car Repair plan item should allow the user to set the estimated cost of the repair. The estimated cost may also be derived from a product provision. A product provision is the delivery of a service by a provider at a specific location. For example, an organization has defined a plan item to provide basic computer skills training. Providers have been registered with the organization that offers this type of training at a number of different locations at different costs. The cost of delivering the service depends upon the provision (a combination of provider and location) selected. Therefore the estimated cost of a plan item added to a service plan to provide this type of service derives its estimated cost from that specified for the product provision.

3.5.2 Actual Cost

The actual cost of a plan item is the amount paid by an organization to deliver the service associated with the plan item. Depending on the business requirements for a plan item of a particular type, the actual cost may be derived from the plan item definition, entered by the user or derived from the payments made in respect of a product delivery.

3.5.2.1 Derived from Plan Item Definition

The actual cost of a plan item may be derived from the cost attribute of the PlanItem entity. Typically the actual cost should be set in this manner when the cost of the service to be delivered is fixed and may not vary, e.g. a bus pass. Therefore the user should not be able to modify the value of the actual cost.

3.5.2.2 User Entered

For certain types of plan item, the actual cost may be set by the user. The actual cost should be set in this manner when the estimated and the actual cost can differ and the actual cost is not derived from elsewhere, e.g. a product delivery.

3.5.2.3 Product Delivery

Certain types of plan item result in the creation of a product delivery to manage the provision of the service to the client, e.g. a Mileage Reimbursement case. The product delivery will use the standard application financial processing to issue payments to the client or the service provider. Therefore the total amount of the payments made on the case at any point in time is equivalent to the actual cost to the organization at that time. The user should therefore not be able to enter the value for the actual cost of the plan item.

3.6 Service Plan and Provider Integration

The addition of a plan item to a service plan can result in the authorization of a client to receive a service from a provider. The 'Service' and 'Custom Service' plan item types enable an agency to authorize the delivery of services to clients. Upon delivery of the service from a provider, information such as service delivery date and number of units delivered is updated on the plan item.

3.7 Integration with the Cúram Provider Management module

Upon approval of a 'Service' or 'Custom Service' type of plan item, the system will invoke the Create Service Authorization API to create a service authorization for the planned item. The Case ID and Case Participant Role ID of the concerning participant in the parent case (e.g. Integrated Case) are used as input parameters when creating the service authorization. The system will then create from one to many service authorization line items for the service authorization depending upon whether or not a frequency has been specified for the plan item.

Units Authorized, Rate Authorized, Expected Start Date, Expected End Date, Provider, and Provider Type are used as input parameters while creating the service authorization line item(s) for the planned item.

Following are the APIs available from the Cúram Provider Management Cúram Provider Management (CPM) module that are used to create and maintain service authorization related information.

Table 3. CPM APIs

Method	Attributes
Create Service Authorization Line Item ServiceAuthorizationLineItem.insertServiceAuthorizationLineItem()	Service Offering Provider Type Provider Units Authorized Start and End Dates MaximumUnits ServiceAuthorization UnitAmount UnitAmountFixed Nominee
Create Service Authorization ServiceAuthorization.insertServiceAuthorization()	CaseParticipantRoleID
Cancel Service Authorization Line item ServiceAuthorizationLineItem.cancelServiceAuthorizationLineItem()	serviceAuthorizationLineItem
Modify Service Authorization Line item ServiceAuthorizationLineItem.modifyServiceAuthorizationLineItem()	modificationReason UnitAmountFixed UnitAmount UnitsAuthorized

Upon delivery of a unit(s) of a service either through invoice processing or attendance tracking, the system will automatically update the Actual Start Date, Actual End Date, Actual Cost, Status and Units Delivered information of a service plan item.

The following events are raised by the CPM module.

Table 4. Events raised by CPM

Event Raised	Parameters	Additional Info
SILIAPPROVED	PrimaryEventData: ServiceInvoiceLineItemID SecondaryEventData: None	* Update Planned Item Details * Update Client Participation
PRLI_APPROVED	PrimaryEventData: ProviderRosterLineItemID SecondaryEventData: None	* Update Attendance Information * Update Planned Item Details
PRLIC_APPROVED	PrimaryEventData: PRLICorrectionID SecondaryEventData: None	* Update Attendance Information * Update Planned Item Details

3.7.1 Update Client Participation:

* Create daily attendance record for planned item, if the service offering is of non placement type and if Attendance Tracking Enabled indicator for the service in CPM is set to No effective from the 'Service Date From' to the 'Service Date To' of the Service Invoice Line Item or

the Attendance Tracking Enabled indicator for the service in CPM is set to Yes and the 'Daily Attendance Tracking Required' indicator for the service in CPM is set to No.

* Update 'Total Hours Participated', if unit of measure of service offering is 'Hours'

Update Planned Item Details:

* Update actual start date, actual end date, actual cost, units delivered and planned item status information.

Update Attendance Information

* Associate Absence and Daily Attendance records to planned item

3.8 New System Properties

The following are the system properties related to the 'service' and 'custom service' type of plan items.

Table 5. System properties for 'service' and 'custom service' type plan items.

Name	Default
curam.serviceplan.serviceplanhook.registrars (Used for validation of ent mods attributes)	This field is used for validation of the Provider Specific attributes and the registration of those hooks.
curam.serviceplans.showResponsibilityOrConcerning	The Gantt chart will also display the concerning participant name or responsible participant name next to the planned item name, depending upon the configuration option chosen. This field needs to be configured to 'Responsibility' or 'Concerning' accordingly.
curam.serviceplan.events.raiseEventOnServicePlanApproval	This field allows the user to set a configuration option for turning ON or OFF the event at the point of approval of a service plan that would enable the agency to add solution specific validations
curam.serviceplans.closepreviousplanonclone	This field enables the user to clone service plan by closing the existing service plan or clone service plan without closing the existing service plan.

Chapter 4. Implementing a Plan Item

4.1 Introduction

The implementation of a plan item varies depending upon the required functionality. However there are some fundamental steps that must be undertaken in order to allow the plan item to function in a service plan. The following example, Mileage Reimbursement, illustrates the steps that must be undertaken to add a plan item to a service plan that results in the creation of a product delivery. This chapter uses the Mileage Reimbursement example to outline the configuration required to achieve the integration between a service plan item and a product delivery case.

4.2 Mileage Reimbursement Example

Social Enterprise agencies provide mileage reimbursement to clients who are participating in scheduled activities as part of a service plan, e.g. job search or job placement. The agency will pay the participant mileage at a specified rate if they use their own vehicle for traveling to a scheduled activity. Most agencies pay mileage at a specified rate up to a maximum amount e.g. \$200 per month. This type of requirement is typically implemented in the application using a product delivery case. Therefore the addition of a Mileage Reimbursement plan item to a service plan should result in the creation of a product delivery. The product delivery is used to pay the client the appropriate amount in respect of the mileage claimed. This document is primarily concerned with the design of the integration between the product delivery case and the service plan; the elements of design specific to the product delivery case (i.e. the evidence and rules) are beyond the scope of this document and are not considered.

4.2.1 Adding a Mileage Reimbursement Plan Item

To recap, the addition of a plan item to a service plan to provide a client with Mileage Reimbursement should result in the creation of a product delivery case to pay the client the appropriate amount. However, at what point should the product delivery be created? If the product delivery is created at the same time as the plan item is added to the service plan, what should happen if the case worker inadvertently added the plan item or a supervisor rejected a request to approve the plan item? If the product delivery had been created, it would have to be closed (a product delivery cannot be deleted). This seems inappropriate. It would be preferable to defer the creation of the product delivery case until the associated plan item (or the service plan) has been approved. The approval of the plan item should therefore trigger the creation of the product delivery. However in order to create the product delivery, mileage reimbursement specific information (e.g. mileage rate and maximum monthly amount) must be captured and stored at the time the plan item is added to the service plan. The `PlannedItem` entity is used to store the generic plan specific information required to add a plan item to a service plan, e.g. expected start date, expected end date etc. However, it cannot be used to store the information required for a specific type of plan item, e.g. in the case of mileage reimbursement, mileage rate and maximum monthly amount. Therefore an entity must be defined to store this information. The combination of this new entity, let's call it `MileageReimbursementData`, and the associated `PlannedItem` entity contain all of the attributes required to create a Mileage Reimbursement product delivery upon approval of a Mileage Reimbursement plan item.

The two entities are linked using the `PlannedItem` entity's associated `ID Number` attribute which acts as a foreign key to the `MileageReimbursementData` entity. In order to populate these entities a client screen and associated server business processing must be developed. The client screen must allow the user to enter the information to be stored on `PlannedItem` and `MileageReimbursementData`; e.g., mileage rate, maximum monthly amount, expected start date, which will be later used to create the Mileage Reimbursement product delivery.

Server business processing must be implemented to insert the plan related information (e.g. expected start date, expected end date etc.) and the Mileage Reimbursement specific information into the appropriate entities. Interfaces are provided for the PlannedItem entity as part of the Cúram Service Planning framework; however custom development is required for any new entities created to store the plan item specific information, e.g. MileageReimbursementData. It is important to note at this stage an associated product delivery case has not been created. Therefore if the case worker deletes the plan item there is no associated case to close. So when does the case get created? In this example, we want the case to be created when the Mileage Reimbursement plan item is approved. A workflow event is raised when a plan item is approved at present. This causes a workflow process to be enacted and this workflow process can be extended to incorporate the creation of the Mileage Reimbursement product delivery. The workflow must be extended to include the processing required to create the appropriate product delivery. An additional transition must be added to the workflow that includes a condition based on the plan item type. If the plan item type equals that defined for a Mileage Reimbursement plan item, then the workflow will follow the path that results in the creation of a Mileage Reimbursement product delivery. To create the Mileage Reimbursement product delivery custom code must be implemented to create, approve and activate the case and finally to insert the value for the mileage rate and maximum monthly amount as case evidence. Note, case evidence must be captured before the case is approved. Once the custom code is available for these steps, the processes can then be automated by adding each process as an automatic step in the workflow. A minimum set of data is required in order to create any product delivery. The following table details the required data and the value that each item is assigned for the mileage reimbursement product delivery.

Table 6. Product Delivery Creation Data

Data Item	Value
Client ID	The plan participant ID. This is the participant ID of the registered client for whom the service plan has been created.
Product ID	This is the ID of the mileage reimbursement product.
Product Provider ID	The mileage reimbursement product is provided by the organization and therefore this can be set to the default organization provider value.
Product Provider Location ID	Same as above, except this should be set to the appropriate default location.
Delivery Pattern	Set to the default that has been defined for the mileage reimbursement product.
Expected Outcome (this was Objective in previous Curam versions)	This should be set to the expected outcome that has been defined for the plan item. In the case of a mileage reimbursement plan item, an expected outcome is not really appropriate (other than it is expected that the client will receive mileage reimbursement payments). Therefore, the user should not be given the option to select this (though they could be) and it should be defaulted to an appropriate value.
Currency	Set to the default organization currency.
Start Date	Set to the value of the expected start date entered for the plan item.
Expected End Date	Set to the value of the expected end date entered for the plan item.
Case Owner	Set to the user selected as the plan owner.

As stated above, on approval of the plan item the associated product delivery case should not only be created, it should also be automatically approved (a supervisor has already approved the plan item and therefore by extension the associated product delivery) and activated. The mileage rate and maximum monthly payment amount should be inserted as case evidence to be used in the payment determination. On creation of the product delivery a ProductDeliveryPlanItemLink record must be written that contains the ID of the product delivery and the associated plan item. This is used to associate the product delivery with the plan item and vice versa. The MileageReimbursementData record associated with the plan item can be deleted at this point as it is no longer required.

4.2.2 Modifying the Mileage Reimbursement Plan Item

The plan item data that can be modified depends upon whether the case has been created and subsequently activated. Upon case activation, the expected start and end dates and the expected outcome may no longer be modified (the plan item is considered In Progress). As we are dealing with a plan item that results in the creation of a product delivery, the actual start and end date and the actual cost must be derived from the product delivery and may not be entered by the user in respect of the plan item. The plan item outcome may not be entered until the associated product delivery is closed (the plan item is Completed).

Because the user should not enter the actual start or end date or the actual cost, the modify page for this type of plan item would not include these fields. Also, the processing associated with the mileage reimbursement specific details - mileage rate and amount, depends upon whether the associated product delivery has been created. If the product delivery has not yet been created, then the value entered for these fields should update the values stored on the MileageReimbursementData entity. However, once the product delivery has been created, depending on the specific requirement updating these fields should either not be allowed or should result in an update to the case evidence (if it is an update to the case evidence, consideration must be given to the date from which this change is effective).

4.2.3 Viewing Mileage Reimbursement Plan Item Details

We have already discussed how the addition of a plan item to a service plan can result in the creation of a product delivery case. However, the integration between the plan item and the case should not end at that point. To provide full integration between the plan item and the product delivery case, the user should be able to maintain the critical aspects of the product delivery case from the plan item. Therefore the view page designed to display the details of the plan item, should also display the most pertinent case information. In addition, the user should be able to maintain this case information from the plan item details page (this same information should also be maintainable on the case).

The most important information on the Mileage Reimbursement product delivery is the distance traveled by the client and therefore the mileage claimed. The addition of this information is also the most common processing that will occur in respect of the case. Therefore, the plan item details page should display this information and provide the user with the ability to enter it. While the details may need to be visible from the plan item, the full case management facilities will always be available. A hyperlink could be added for easy access to the Mileage Reimbursement product delivery. This page may be opened by the user before the case has been created, i.e. the plan item (or service plan) has not been approved. In this instance there will be no case related information to display. An informational could be returned to inform the user that the case has not yet been created.

Chapter 5. Compliancy

5.1 Miscellaneous Entities

As part of V6 application development, Service Plans created following new entities in Service Plans component.

- Goodcauselink
- Outcomelink

These entities were created to make linking of Outcomes and Good Causes in service planning to be generic. If required, customers can customize the same.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service. IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law.

IBM Japan Ltd.

19-21, Nihonbashi-Hakozakicho, Chuo-ku

Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you. Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

Dept F6, Bldg 1

294 Route 100

Somers NY 10589-3216

U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources.

IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing

application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/us/en/copytrade.shtml>.

Java and all Java-based trademarks and logos are registered trademarks of Oracle and/or its affiliates.

Other names may be trademarks of their respective owners. Other company, product, and service names may be trademarks or service marks of others.



Printed in USA