Program Management in WebCenter

Technical White Paper

Shalet K. Varghese

March 2017
1 Introduction..................................................................................................................................................................... 3
2 Essential Packaging Program Management Features...................................................................................................... 3
  2.1 Briefing and Approval ............................................................................................................................................... 4
  2.2 Adding projects to a Program ................................................................................................................................... 4
  2.3 Program Manager's Dashboard ................................................................................................................................ 4
  2.4 Metadata sharing ...................................................................................................................................................... 5
  2.5 Mutual Communication ............................................................................................................................................ 5
3 The WebCenter Solution ................................................................................................................................................. 5
  3.1 Briefing: Definition and Approval of Components.................................................................................................... 7
    3.1.1 Defining Products and Components ....................................................................................................................... 7
    3.1.2 Approval of the Products and Components ........................................................................................................... 8
    3.1.3 Relationship between the Products and Components ........................................................................................... 8
  3.2 Creating a Program ................................................................................................................................................... 8
    3.2.1 Adding/Creating projects in a Program .................................................................................................................. 9
  3.3 Metadata / Attributes ............................................................................................................................................. 11
  3.4 Program Overviews ................................................................................................................................................. 11
    3.4.1 Counters in a Dashboard ...................................................................................................................................... 11
    3.4.2 Charts to visualize your program data .................................................................................................................. 12
    3.4.3 Drilldown .............................................................................................................................................................. 12
    3.4.4 Actions from your Dashboard ............................................................................................................................... 13
  3.5 Overview inside the Programs and projects ........................................................................................................... 13
  3.6 Communication Channels ....................................................................................................................................... 14
    3.6.1 Configuring Communication Channels in Programs and projects ........................................................................ 14
4 Conclusion ..................................................................................................................................................................... 15
As marketing and packaging programs become more complex and distributed, the need to automate and manage them for efficiency, transparency and risk management is clear. WebCenter program management is a unique system that manages both marketing and packaging programs and supports the automation of the overall process involved in the development of these assets. The system supports both complex and basic programs with configurable settings, allowing each organization to create a platform for their unique needs. This paper reviews how WebCenter supports the creation and structuring of programs, the development and processing of deliverables and activities, as well as how program reporting is handled.

1 Introduction

Brand Owners are constantly confronted with many packaging programs or campaigns which are triggered by new product introduction to market, promotions and regulatory updates. They are tackling large, complex packaging programs (campaigns) which span across packaging ideation to the delivery of individual packaging to specific markets. Typically these efforts involve several parallel and sequential projects. While a packaging project involves dealing with a relatively short-term objective, a packaging program is much more complex and deals with several short-term and long-term objectives and requires a large pool of multi-disciplinary resources.

Accomplishing a program’s objectives involve planning and conducting a number of related projects, multiple work streams with their own requirements and planning, the results of which are components of the overall outcome. Therefore, overall packaging management involves taking control of such programs with a common leadership which is a step further than what is defined in traditional project management approaches. When you have large packaging programs with multiple managers who are responsible for planning, control, and updating activities, your project management platform should provide easy visibility of both the program and project level view on progress and risk.

This white paper provides an overview of the approaches and characteristics of Program Management in WebCenter. This document is designed for Esko’s internal audience who already work with WebCenter project management workspace and intends to provide a starting point to the WebCenter Program Management module.

2 Essential Packaging Program Management Features

This section discusses the fundamentals of a solution to address this complex scenario. The approach described here is based on the experience of deploying a packaging program management solution at customer premises. From a brand owner’s perspective, to manage the campaigns effectively, there should be a way to group them together under a program, yet conduct them individually balancing the needs of the program and that of the projects. In the case of packaging campaigns, it could be that some projects contribute to the outcome of more than one program.

The foundational feature that is a must is an overarching program which describes the project outline, rationale, launch criteria, dates, regions, and suppliers which would also always keep its link with its active sub-projects.

An ideal solution will enable you to:

- Break your program into smaller projects
- View and track the program’s progress with a dashboard type view
- Streamline the briefing and approval (of components) process
- Automate the project creation
- Track and manage deliverables of each project (for the allocated budget) individually
- Manage schedules (of projects and the program)
- Reuse existing projects and components in a program
Program Management in WebCenter

- Share metadata within a program
- Define metadata, milestones, and requirements at the start of a program
- Host packaging components in individual projects where you can
- Manage document uploads and approvals
- Navigate the program and project hierarchy

2.1 Briefing and Approval

Typically, to initiate a new campaign, the Brand Manager starts up with the briefing process based on the marketing requirements. During this phase, he or she should be able to define the existing and new components very easily and bring them on to the briefing and approval process flows. Ideally, it should be possible to define the components and send them through the approval process simultaneously.

2.2 Adding projects to a Program

While an automated process is desirable in some cases, as a program owner, you may want to create or add projects to the campaign manually. Sometimes, it may be possible to reuse a project in multiple programs. This warrants a solution which has the ability to create projects manually and reuse existing projects in a program structure.

After the projects are created in a program, they can have their own life within the program. Further, these projects may have child projects depending on the complexity of its objectives. Therefore, the solution should support multiple levels of projects (with managers, project initiators, Packaging Managers, and Marketing Managers at these levels) where projects can be added manually (depending on their permissions) or automatically.

When projects are created in a program, they should exist as "related". That is, there is some linking bond, contents, and objectives they share; and which excludes other projects in the system that do not share this connection. If you select unlinked projects and create a campaign, there should be a way to create this link before declaring them as part of the program. These connections build a hierarchical structure of parent and child projects that should be easily traversed.

2.3 Program Manager's Dashboard

Once the program is started, the brand manager or the program manager should be able to track and monitor the diagnostics of a program with a simple overview. For example, the program manager can easily pull up a list of projects from a single campaign and quickly generate a status report. Yet another scenario is to find all the components of a particular brand campaign.

For easy tracking, the program manager’s dashboard should indicate and display the statuses of its sub-projects. At any point of time, a program manager should be able to:

- See a quick overview of the campaign projects where you can refine the view based on some criteria
- Check on the status and duration of the sub-projects or track the milestones (with status indicator) and diagnose problems
- Identify potential risks in the sub-projects and interdependencies. Ability to focus on problem area
- Drill-down to child projects, which lets you explore project data beyond the data immediately visible on the dashboard. If your program has different levels, for example, Project / Grouping / SKU / Components, then drilling up and down is essential to have a complete picture.
2.4 Metadata sharing

When projects are created in or added to a program, they would ideally share some of the metadata. The inter-project and project-program dependencies should be captured in shared metadata, inheritable and updatable from each level of the program hierarchy. Therefore, the solution should have features to:

- Propagate data from the program to child projects when they are being created. The program should propagate the shared attribute values automatically when a child project is created (or added) in a program.
- A change in program’s value should update values in its member projects
- Copy metadata when new member projects are created from a child project

2.5 Mutual Communication

After a program’s launch, an effective communication line should be always available between the projects and the program, regarding important events such as milestone date change, project status change, etc. For example, if all projects of a program phase are completed, the program’s status should be updated as completed. In the reverse scenario, if one of your projects is in delay, the program status should also indicate the risk. Yet another example would be, to alert the program of important and critical events happening in the child projects. Further, when a Project Manager of a child project decides to alter the schedule, this should be alerted to the top level and related projects. Typically, program schedule should spill down to the projects and schedule changes from projects should flow upwards to reflect on the program level.

3 The WebCenter Solution

WebCenter Program Management offers features to manage programs, projects, and deliverables in a hierarchical structure. You can specify components, create a group of projects in a program and modify them later if needed. In addition, the program manager has the full visibility of the program, projects, and components and an extensive dashboard can reflect on the diagnostics of a particular program (or of all programs). The program hierarchy can have shared attributes such as milestones and common project members. Further, a program can update its child projects of an event and vice versa, and these communication channels help you manage risk and resources among the member projects of a program.

The solution offers visibility and communication from all levels. However, when you need restricted visibility, you can alter the security settings to restrict access. WebCenter Program management offers:

- Component definition and approval
- Creation of program’s projects
- Seamless communication between the program and its projects.
- Ability to attach (and remove) existing projects and components into a program
- Features to support multiple levels: for example, a Project/Grouping/SKU/Component.
- Shared projects and components among different programs (one project can have multiple parent projects)
- A program overview with drill-down where you can monitor statuses, due dates, different phases etc.
3.1 Briefing: Definition and Approval of Components

Program planning may require multiple iterations and produce multiple work products and components. WebCenter has the ability to deal with this multiplicity with its feature set. We envision that you can create the planning effort in two parts. The first deals with the overarching program setup which describes the outline, rationale, launch criteria, important dates and involved regions and suppliers. The second part deals with the individual or child projects within the program which is entirely dependent on the product and component planning. Further, based on the defined products and components, spawn off the creation of projects either manually or in an automated way.

The program defines the connections and dependencies for the component projects and lets you add or link more projects into the program framework in phases. This allows enough flexibility to refine and rework their project plans to integrate with the program. For example, when a new marketing campaign is initiated, the packaging manager often has to execute a set of projects as a program. This starts with the briefing phase to define the components. It can be completely new components or components from repositories with the proposed changes. See details in the following sections.

3.1.1 Defining Products and Components

To start a program with new products and components, the Program Manager can use a placeholder document (Document Templates). These templates serve as a starting point for actual documents (with all the pre-defined metadata and approval setup), which gets replaced with the right document during the later phases of the Program. To define, existing components just refer to them (document references) or copy them to the current scenario. You can source the components from existing libraries, contextual saved searches (SmartName powered), or from a previous program. The system also allows you to duplicate such components (with all its metadata and related documents) to create variants. This way, you can first define your product-component structure to be further processed as individual (or grouped) projects. See the example below. The following structure can be built into your Program Template making it easy for the Program Manager to source the different components.

As an example, a packaging program can be initiated for multiple SKUs for one product. When a program manager launches the program with the SKU templates, the Artwork Project Managers can be assigned to verify and add additional SKUs if needed. Once these SKUs are approved (through an approval cycle), the Program Manager can track them as SKU-projects to which Packaging Engineers will add all the relevant components. Further, each SKU needs packaging content and translations which can also be tracked as a separate Packaging Content Management project if
desired. Note that the SKU-projects may be shared across different programs, components can be shared across SKU-projects and each component can have its own pre-press resources. You can see an example of a workflow in the screenshot.

3.1.2 Approval of the Products and Components

Once you have defined your products and components, you have the options to start the Approval (simple or staged) process for each of those components unless you are sourcing already approved components. It is also possible to automate this process with a pre-defined workflow to be launched at the creation of the program (with specified components) or manually by the Project Managers (see the workflow from the previous screenshot). Depending upon your business logic, it may suit you to launch the program based on approved components and add or link more projects later on. The features of the solution also allow you to execute your program in phases or waves by letting you add your products, components, and projects even after the program launch. See details in Adding/Creating projects in a Program.

3.1.3 Relationship between the Products and Components

You can make use of the Document Referencing feature to specify product and its components or parent-child relationships of documents. In addition to the Program-Project relationship, the document referencing feature helps you in organizing the product-component relationship. This relationship keeps the product and components together even if they are stored in different repositories. This way, the products can keep common components along with their own metadata thus allowing you to use the already created and approved parts thus allowing you to move to production quickly. You have the ability to link already approved components and request new components for a new product.

3.2 Creating a Program

The WebCenter Program Management module allows several projects to belong together for tracking purposes (for example, a single customer order can be a master project with a list of child projects). The projects created under the hood of a program and those added on to the program remain together with open communication channels. Further, the solution offers a drill-down view for the Program Managers for tracking and monitoring the individual programs and child projects. See more in Program Dashboard with a Drill-down View.
A program manager can create or start such a program with a simple ‘Create’ or a ‘New Program’ action. The action will use a Program Template with all the pre-configured program data such as the schedule, members, and security settings. The managers (administrators) can secure the programs and projects by setting permissions for users and groups at the template level. For example, you can decide what a designer accessing a component project can view. It is also possible to apply these permissions on folders within the project.

You can add or create projects in a program in a few ways which offers extensive configurability and flexibility. These are all configurable actions which means that you can include the relevant actions depending on your user type. All child projects created or added in program shares attribute values, role assignment and maintain an active link with the program. The child project creation (or addition) is achieved through configurable action which means that your administrator can retain only those actions that make sense in your business context.

### 3.2.1 Adding/Creating projects in a Program

When the components are approved, you can create child projects as members of the program with shared attribute values and company information. It is possible to create projects and manage them in multiple tiers. In a program containing multiple SKU-related projects, each project may drive its own component and content projects.

#### 3.2.1.1 Manual Project Creation (From a Program)

The Program Manager can create new child projects individually with pre-defined custom child project Template. This allows you to add projects to a program using a Create Child action from the context of a program.

This provides an interactive way of creating a child project where the program’s attributes are inherited (for example, milestones set in the program and Role setup). The child project can be created with its own attributes and documents apart from what is propagated from the program. Also, you can have a different thumbnail, planning, workflows, folder structure, approval cycle, and security settings in these child projects. All these can be configured in a dedicated Template that is attached to the Create child project action. It is possible to select components or templates for components manually during project creation. To make this possible, configure contextual document sources using SmartNames. SmartNames in document sources allows you to select from the components that are appropriate for the context. You can also use a workflow to select the right components in order to launch them in the child project.
3.2.1.2 Automated Project Creation (From a Program)
You can automate the creation of child projects if you have structured your content while defining your program. You can either launch this automatic action after the start of the program or launch a workflow action from the program’s detail page. See the screenshot below to see an example of auto-creation of projects from a program. This action takes a pre-configured workflow created based on certain attributes (for example, region) and structured content to create child projects in one go. This workflow can be configured with a Create Project node and the workflow logic can be determined by your administrator.

This method is useful for programs with several SKUs each of which to be managed in an individual child project. Another use case would be to launch your program in different phases or waves. To create your projects in waves, you would launch this action periodically, for example, when the previous wave has been successfully completed.

3.2.1.3 Add an Existing Project to a Program
In situations where you can reuse existing projects, you can add them to a specific program. The module allows sharing of a project within several programs. For example, you can add an approved SKU to several programs instead of creating another project. You can use anAttach action to select and add an existing project to the program. You can use a contextual Search (for example, based on common attribute values or SmartNames) in theAttach action. This way, you can narrow your “Select project” pop-up dialog to display the most relevant projects to choose from. See an example of such a pop-up below.

3.2.1.4 Create a new Project from an Existing Project
You can create child projects based on existing projects by using a ‘Create sibling’ action from within a child. The ‘Create sibling’ action allows you to create a clone of that particular project. For instance, after creating component projects, you may see that you need an extra component with the same requirements. When you create another child project from an existing one, all the project’s settings are copied to the new one by default. However, the manager can decide to eliminate the items that are not needed. Individual project Managers (or Program Manager) can always add new items to an existing child project.

3.2.1.5 Delete or Unlink Project from a Program
As a Program Manager, you also want features to remove a project from a program (without deleting it) and to delete a project from the system. You can configure and use Unlink and Delete actions in a program. All of these actions are con-
figurable and you can opt for certain actions or all of them depending on your requirements. Read more in the WebCenter Documentation.

3.3 Metadata / Attributes

The program and its child projects are linked together in a way that it is possible to navigate between parent and child and also share some common attributes based on the template configuration. As described earlier, the attributes that are to be shared can be configured in the Templates of the program and the child project. When a child project is created in the context of a program, the child project gets the value of the attribute from the parent. For example, company information, specifications, role assignment or milestones. You can determine which attributes are to be copied to the child projects from the program based on your needs. The child projects and program projects can have attributes that are not shared amongst them.

You can also use attributes in the Program to store and reflect KPIs. For example, you can use attributes to store the status of the program which in turn would need to recalculated based on child projects’ events and status updates. During the configuration, the administrators can place the KPI indicators in the program and the child project and use a communication channel to keep their values up to date.

3.4 Program Overviews

The Program Management solution offers different types of overviews of on-going programs as a way to easily monitor your program and programs. This section details the high-level drill-down overview displaying your ongoing programs along with their individual summary and status. This overview is configurable and you can use most relevant attributes as the criteria for summary and status traffic lights. The example screenshot gives an overview where the Program Manager can quickly zoom into programs based on their status (which in turn is based on the status of their child projects). In addition to this bird’s eye view, the drill-down feature also allows you to quickly access the specific projects or components that require special attention.

3.4.1 Counters in a Dashboard

Counters can be used to give a high-level diagnostic overview of the status of your program. You can, for example, show the project Statuses in these traffic light counters (at risk, delayed, or on track). With a simple click on one of these counters, you can zoom into the relevant projects. For instance, a click on the “On Track” counter will give you a list of the child projects which are ‘On Track’. Like with status, any project attribute can be monitored using counters.
3.4.2 Charts to visualize your program data

Like the counters, you can use any of the project Attributes to configure these charts/graphs. The fields in the graph show the distribution of values and can be used to dig deeper. If you click on a facet, it will refine the results to match the facet. This can be used in any or all of the program levels. In the screenshot, the counters provide the most high-level overview whereas the first drill-down displays a pie chart. Your pie charts/donut charts or bar graphs can be configured at any level in the hierarchy and can be used as a visual diagnostic.

The charts allow you to summarize your program data. For example, you can add a chart to reflect on your program phases.

3.4.3 Drilldown

You can enhance your overview with a drill-down list of all programs or projects in a program depending on which level you are. Drill-down into all levels of projects allows the Program Manager to explore the project data beyond the overview. If you have a nested structure (e.g. Project / Grouping / SKU / components) in your program, an expand feature lets you access each level in an easy way. The manager can, if needed, zoom into potential risks arising from the child projects and their mutual dependencies.
You can configure this overview to further drill-down into your component structure as well. In the example given here, the Program Manager drills down into the ‘Overdue’ program and digs deeper into the SKU and component levels respectively.

### 3.4.4 Actions from your Dashboard

A Program Manager can execute Actions from the program overview. In the example given below, to launch a briefing workflow from the overview page, the manager can select a few items and click the ‘Start Briefing’ workflow action. Note that these actions are configurable and the workflows you want to run behind these actions can be configured to reflect your business processes.

### 3.5 Overview inside the Programs and projects

For ease of access, our solution offers complete visibility from either side of the program. This makes the child projects and programs accessible from both directions. If you are a Program Manager working on a Program’s details page, you have direct access to that program’s projects. Likewise, from each of the child project, she or he can jump right back into
the parent or program.

3.6 Communication Channels

Since the projects are children or sub-projects of the overall program, there is a need for communication among all levels of the hierarchy, program-to-projects, projects-to-program, and among projects. In a program setup, it is imperative that there is a live communication channel between the projects and the program.

The updates happening in a program have mutual consequences and therefore should be communicated in both ways. If a project manager performs changes on a subproject, this must be communicated to the overall program dashboard. Conversely, if the program manager updates something in the master project those changes will also appear in all the affected sub-projects.

The WebCenter solution provides the ability to:

- Alert the parent or a related project about a new deliverable
- Alter the status of a related project depending on another
- Notify program when a child project encounters change: a status update or metadata change.

3.6.1 Configuring Communication Channels in Programs and projects

The Administrator of the system can configure "actions" "triggered" by an "event". These are Rules that can be specified in the program/project’s details. You can set these up based on a program’s specifications and milestones using a JavaScript workflow driven action.

The triggered actions will be implemented in the program or project specified in the attached "action". JavaScript powered WebCenter workflows run in the background to execute these actions and all the changes made by these "actions" will be stored in the history as part of the audit trail. The scripting capabilities embedded in this feature make this a flexible and configurable tool which can accommodate complex business logic.

The configuration of the Rules is based on Program/Project Template and therefore, once setup, can be reused in future programs. It is also possible to export and import your established communication channels.

See some examples below:

- If a child project’s status changes to Overdue, change the status of the program to At Risk
- Change attribute values in a program based on the values in the child project
4 Conclusion

Packaging is becoming more complex, customized and unique, with increasing regulatory and market pressures all of which make it a challenge to overcome for both Brand Owners and for all elements in the supply chain – from concept to print. Program Management helps all stakeholders to know how, what and when to perform certain tasks – while also promoting automation – to achieve the short and long term objectives of a new packaging-related project.

As described in this white paper, WebCenter Program Management is a solution that supports the differentiation and automation of different projects, while being attached to a global Program. It helps Brand, Program, and Project managers to track and manage all deliverables, schedules, status, and all assets involved in the program while also providing an easy to use dashboard overview.

From the Marketing Department at a Brand Owner to the Printer / Converter, from concept to print, WebCenter Program Management increases visibility, traceability and overall efficiency of the whole process helping them to face the new packaging project related complexities.