What are advantages of SAP Business Workflows?

A business process can consist of several steps. Historically, the tasks have been coordinated manually or by some informal means of communication (sticky note, e-mail, shouting, and so on).

The common problem in these approaches is:

- Inefficiency
- Each lacks a way to trace where a task is
- Who executed (or is executing it),
- How much time it requires.

In contrast, the workflow ensures that the right work is sent to the right person at the right time in the right sequence with the right information.

- Workflow is a tool for automation of business processes
- Not tied up to any particular application
- Operates uniformly across applications
- Coordinate all participating tasks
- Provide users with active support
- Deadline Management is possible

Workflows are very useful for business processes that:

- Contain a sequence of activities.
- Reoccur in the same or similar forms.
- Involve several people or groups of people.
- Require considerable coordination.

What are the layers in Workflow Architecture?

There are 3 layers in the Workflow Architecture:

- Business Object
- Business Process
- Organization Model

How Process, Business Logic and People are linked in Workflows?

- The business processes are defined in Workflow builder. Each and every small work process is called as an activity in the workflow builder. Each activity consists of a single step task, which can be a standard task or a workflow template.
- Associated Business Logic is built in Business Objects. i.e. ABAP code will be in Business Object. Business Objects consist of attributes, methods and events. Methods consist of ABAP code. Each and every standard task is associated with a Business Object and a method.
- People / Group of people responsible for taking actions are defined in organization structure. Person responsible for taking an action is called actual agent. It is defined in the workflow activity.

What is a Business Object?
Business object (BO) is basically a collection of Attributes, Methods and Events for a particular business entity. Example of BO: Sale Order, Vendor, Customer, Material, etc. BOS wrap around backend tables, application code, change document and other technical information and expose them as Attributes, Methods and Events. Usually methods of BOS refer to BAPIs.

The following diagram explains BO much better:

What is an Agent and what are different types of Agent?

An agent is a person or a group of persons who can take an action during workflow runtime. Action can be approval of some business document, creation of some document or sending email.

There are basically three types of Agents:

- Actual Agent: One who actually takes the action. This is provided in the workflow Activity.
- Possible Agent: All possible persons/entities that can execute a task during runtime. Defines either in Org Structure or task.
- Excluded Agents: Persons who are excluded from taking action on a task. These are assigned in the workflow definition at the workflow activity.

System checks Actual Agent defined in workflow runtime with the Possible Agents defined in the task and the Excluded Agents. Actual Agent should be a subset of Possible Agents minus the Excluded Agents, else Agent resolution fails.

What are the capabilities of Workflow Builder?

What are Events?

Events are actions that occur in the system and that acts as trigger point for any workflow. Eg. Sales order creation is an event.

Eg: Consider a scenario where any sales order after creation subject to approval from superior. So here a workflow comes into play. The workflow needs to start whenever a Sales order is created. The business object BUS2032 is associated to sales order creation. It has event CREATED. So the trigger point or "START EVENTS" in the workflow is the BUS2032's event CREATED.

Then bind the event to the workflow so the data flows from event to the Workflow. e.g.: Sales order is created. So for the workflow to proceed it'll need sales order details. With binding the respective values is mapped to the workflow container.

Then activate the linkage once you are through the workflow creation. This is important as after this only the workflow will listen for the event, in the sense that after this only the workflow is triggered on the event. This is critical also as each active link adds to system load considerably and too many workflows in the test environment can bring the system down. So careful. Deactivate once test is over and workflow is no longer used.
What are the various methods of creating Events?

The various methods of creating Events are:-

- Function module - event creation by calling function module SWE_EVENT_CREATE or SAP_WAPI_CREATE_EVENT.
- Change documents
- Event creation when change documents are written.
- General status management
- Logistics Information System (LIS)
- Event creation upon status changes.
- Business Transaction Events
- Message Control

How can an Event be linked with a Workflow?

Consider the start of any workflow. The START EVENTS tab is taken through the header. The Business Object is specified and binding is generated to the workflow to receive the values and then the link is activated. See the pic below for the Sales Order creation as start event for a workflow.

What are containers, Binding and Binding Editor?

Containers are nothing but the Variables in the Workflow that hold the values in the workflow. They can be simple variables or structures or even objects of any class. So they are best treated as the variables. Binding as explained above is passing the values from events, methods and other externals sources to the container. When an event is binded to the workflow then the data automatically flows from the Event to the Container in the workflow. Binding editor is there as automatic binding may not always pick up the right or desired binding or precisely the mapping generated may not be appropriate or as required due to technical reasons which any techie can guess. So editor gives an option to the developer to edit or generate manually.

What are important Transactions and Reports?

The area menu ‘SWLD’ lists all the transactions associated with Business workflows.
SWDD - Main Workflow Builder.
PFTC - General task maintenance.
SWO1 - Business Object Builder.
SBWP - The Business Workplace SWDM - Business Workflow Explorer
SWELS - Switch Event Trace On/Off
SWETYPV - Display/Maint. Event Type Linkages
SWUS - Test Workflow
SWUI - Start Workflow
SWI6 - Workflow for Objects
SW14 - Workflows for Objects Types
SWDB - Workflow Builder - Create Workflow
SWLV - Maintain Views

**TASK/TASK GROUPS**
PFTC_INS - Create
PFTC_CHG - Change
PFTC_DIS - Display
PFTC_COP - Copy
PFTC_DEL - Delete

**Tcodes Related To Events**

*Events Creation*
BSVW - Status management
NACE - Message Control
AWUW - Logistics Information System (LIS)

*Event Linkages*
SWB_COND - Workflow Start Conditions

**Work item analysis**
SWI2_FREQ - Work Items Per Task
SWI2_DEAD - Work Items With Monitored Deadlines
SWI2_DURA - Work Items By Processing Duration

**Task analysis**
SWI11 - Tasks in Workflows
SWI13 - Task profile

**List of Important Transaction Codes**

For more SAP Business Workflow Transactions, check the link:  

**Where can I find more details about workflows?**

SAP Help - Business Workflows
SDN Forums
Exception in binding of Task and the Method?

It is observed that when we bind a Task with any BO Method then all the fields defined for binding are wiped out after we save the binding i.e. if we want to see the fields taking part in binding of the TASK & METHOD or while binding the workflow and the TASK it is observed that the container elements and task container elements disappear even though binding is established. This happens because of inconsistent buffer synchronization.

To overcome this issue follow the steps in SWDD.

Menu->Extras->Transport->Synchronize runtime buffer, or go to TCODE SWU_OBUF of buffer synchronization.
Basic Function Modules in Workflows?

- **SWW WI_POPUP_FOR_COMPLETION**: shows the completed status of a workflow
- **SWW ADDR_KEY_CREATE_FROM_USER**: display the address of the user or recipient
- **SWW WI_DELETE**: delete work item
- **SWW WI_CREATE_VIA_EVENT**: create work item as result of event
- **SWW WI_DEADLINE_CHANGE**: change of deadline attribute of work item
- **SWW WI_COMP_EVENT_RECEIVE**: complete workitem using events
- **SWW WI_REJECT**: reject work item
- **SWW WI_EVENT_RECEIVE**: retrieve event for a workflow
- **SWW WI_DISABLE**: lock workitem against execution
- **SWW WI_ENABLE**: remove lock against execution of workitem