

Produzione - patent-default-flow: Flusso Proprieta' intellettuali

- [Grafico del flusso](#)
- [Descrizione](#)
- [Etichette Stati](#)
- [Modello Dati](#)
- [Configurazioni](#)
- [Permessi](#)
- [Validazioni](#)
- [Logiche \(action/start\)](#)

Grafico del flusso

**TEAM VISIONE
COMPLETA
(HD)**

**TEAM VISIONE PER
STRUTTURA
(OD)**

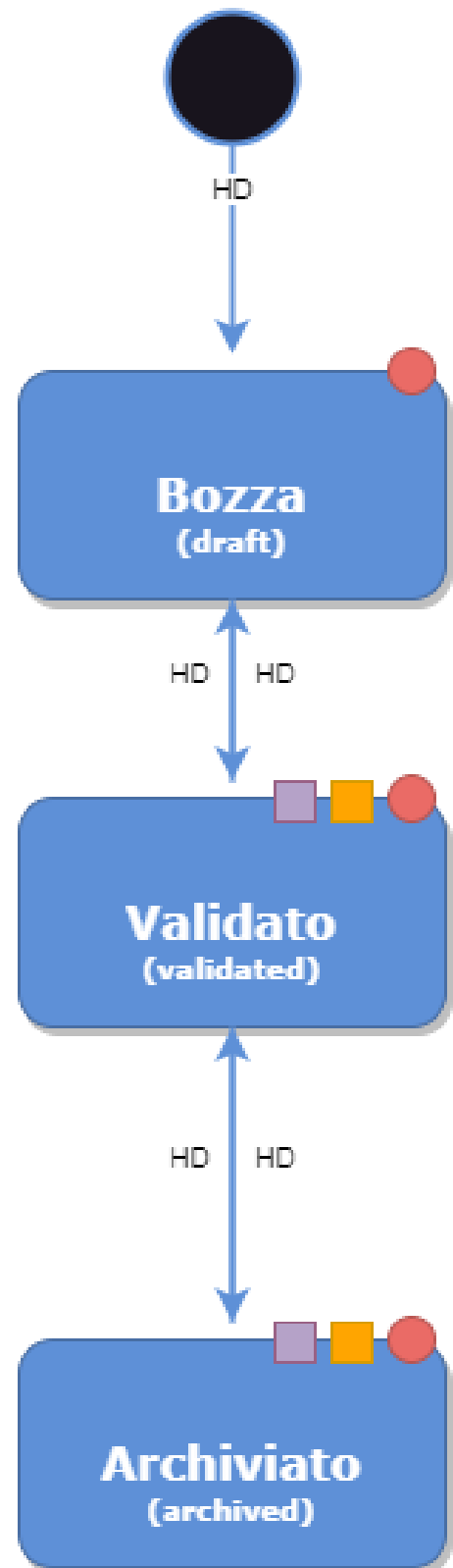
**RESPONSABILE O
PROPRIETARIO
(RS)**

**PARTECIPANTE O
TUTOR
(PA/TU)**

**RESPONSABILE
AMMINISTRATIVO
(RA)**

**DELEGATO O RAPPR.
INTERNO
(RD)**

- read
- ◆ read/write
- read/write/delete



Descrizione

Il flusso patent-default-flow è un **flusso con validazione** per il censimento di una **Proprietà intellettuale**.

Questo flusso prevede i seguenti attori nelle diverse visioni.

- Visione completa
 - **Helpdesk - HD**
Team con profilo "Profilo Helpdesk per le Proprietà intellettuali (patent)"
- Visione dipartimentale
 - **Organi dipartimentali - OD**
Utenti appartenenti al team con la seguente naming convention "Profilo di dipartimento per XXX", dove XXX è il nome del dipartimento
Sono abilitati a questa visione tutti i dipartimenti presenti nella sezione "Dipartimenti di afferenza"
- Visione personale
 - **Responsabili scientifici - RS**
Persona appartenente al team "Utenti", che compare tra i responsabili della Proprietà intellettuale.

Etichette Stati

I nomi degli stati possono essere personalizzati con la funzione di **Gestione etichette**.

Nella tabella sottostante vengono riportati tutti gli stati previsti dal flusso con relativa etichetta e valore di default.

Vengono anche fornite le etichette per i vari bottoni che consentono lo spostamento di stato.

Di default vengono utilizzate le etichette dei bottoni per lo stato successivo, è possibile configurare il flusso in maniera tale da utilizzare anche le etichette dei bottoni per lo stato precedente.

Esempio di utilizzo: l'oggetto si trova nello stato in attesa di essere validato, i possibili stati di cambio sono bozza e operativo. Il bottone per lo stato operativo utilizzerà l'etichetta per lo stato successivo, mentre il bottone per lo stato bozza utilizzerà l'etichetta per lo stato precedente.

Identificativo stato	Chiave etichetta stato	Valore default stato	Etichetta del bottone stato successivo	Etichetta del bottone stato precedente	Valore etichetta del bottone stato successivo	Valore etichetta del bottone stato precedente
draft	wfState.pat.draft	Bozza	button.forward.to.pat.draft	button.backward.to.pat.draft	Salva e invia in "Bozza"	Salva e invia in "Bozza"
validated	wfState.pat.validated	Validato	button.forward.to.pat.validated	button.backward.to.pat.validated	Salva e invia in "Validato"	Salva e invia in "Validato"
archived	wfState.pat.archived	Archiviato	button.forward.to.pat.archived	button.backward.to.pat.archived	Salva e invia in "Archiviato"	@button.backward.to.pat.archived@

Modello Dati

Il dettaglio del modello dati associato a questo flusso è disponibile alla seguente pagina [Produzione - Modello dati \(patent - PAT\)](#)

Configurazioni

Le configurazioni associate a questo flusso sono disponibili alla seguente pagina: [Produzione - IRIS AP-RM Configurazioni: Brevetti \(patent - PAT\)](#)

Permessi

Nella seguente sezione viene riportato il dettaglio dei permessi e delle transizioni di stato possibili per tutti gli attori del flusso.

Per quanto riguarda le transizioni di stato vengono riportati gli identificativi degli stati verso i quali è possibile effettuare la transizione.

Quando viene riportato il marcatore "__PREVIOUS_STATE__" significa che la transizione di stato è consentita verso lo stato precedente.

Di seguito la legenda dei permessi:

- c: create (disponibile solo per il primo stato del flusso)
- r: read
- w: write
- d: delete
- f: forward

Viene, inoltre, fornito dettaglio dei TAB disponibili (quelli in sola lettura presentano il suffisso readonly)

Stato	Attori	Permessi	Transizioni	Tab
-------	--------	----------	-------------	-----

Bozza (draft)	Helpdesk (helpdesk)	c r w d f	validated	- Dati Generali (patent/form1) - Referenti (patent/form2) - Famiglia (patent/form3) - Altre informazioni (patent/form4) - Approvazioni e Scadenze (patent/form5) - Costi (patent/form6) - Parole chiave (patent/form7) - Pubblicazioni e Collegamenti (patent/form8) - Allegati (patent/form9)
Validato (validated)	Helpdesk (helpdesk)	r w d f	draft,archived	- Dati Generali (patent/form1) - Referenti (patent/form2) - Famiglia (patent/form3) - Altre informazioni (patent/form4) - Approvazioni e Scadenze (patent/form5) - Costi (patent/form6) - Parole chiave (patent/form7) - Pubblicazioni e Collegamenti (patent/form8) - Allegati (patent/form9)
	Organi dipartimentali (department)	r	none	
	Responsabile/Proprietario (owner)	r	none	
Archiviato (archived)	Helpdesk (helpdesk)	r w d f	validated	- Dati Generali (patent/form1) - Referenti (patent/form2) - Famiglia (patent/form3) - Altre informazioni (patent/form4) - Approvazioni e Scadenze (patent/form5) - Costi (patent/form6) - Parole chiave (patent/form7) - Pubblicazioni e Collegamenti (patent/form8) - Allegati (patent/form9)
	Organi dipartimentali (department)	r	none	- Dati Generali (patent/form1) - Referenti (patent/form2) - Famiglia (patent/form3) - Altre informazioni (patent/form4) - Approvazioni e Scadenze (patent/form5) - Costi (patent/form6) - Parole chiave (patent/form7) - Pubblicazioni e Collegamenti (patent/form8) - Allegati (patent/form9)
	Responsabile/Proprietario (owner)	r	none	- Dati Generali (patent/form1) - Referenti (patent/form2) - Famiglia (patent/form3) - Altre informazioni (patent/form4) - Approvazioni e Scadenze (patent/form5) - Costi (patent/form6) - Parole chiave (patent/form7) - Pubblicazioni e Collegamenti (patent/form8) - Allegati (patent/form9)

Validazioni

Nella seguente sezione viene riportato il dettaglio delle validazioni per tutte le coppie (attore, stato) del flusso. Le validazioni sono distinte nei seguenti macrotipi e sono riferite, se non specificato altrimenti, all'oggetto radice.

- enter**: validazione applicata in ingresso nello stato
La transizione in ingresso viene NEGATA se anche solo una validazione NON viene superata con successo.
- save**: validazione applicata ad ogni salvataggio e quindi anche per ogni spostamento di TAB
Il salvataggio viene NEGATO se anche solo una validazione NON viene superata con successo.
- delete**: validazione applicata in fase di eliminazione di un oggetto radice
- element**: validazione applicata agli elementi figli di un oggetto radice
- permissions**: logiche di generazione dinamica dei permessi (rwfd) sull'oggetto radice che sovrascrive i permessi di flusso (rwfd)

Le validazioni sono ulteriormente distinte nei seguenti tipi.

- required**: validazione di obbligatorietà di un attributo sull'oggetto radice.
- complex**: validazione complessa applicabile sia all'oggetto radice che agli elementi.
Per avere maggiori dettagli sulla validazione cliccare sull'identificativo della validazione

Nel caso di validazioni di tipo **element**, oltre all'identificativo della validazione, viene riportato anche l'identificativo dell'elemento a cui è applicata e l'azione che l'ha scatenata:

- salvataggio (save)
- eliminazione (delete)

Ad esempio la seguente stringa **internalOrganizationUnit:delete departmentDeleteValidator** indica che la validazione "departmentDeleteValidator" è applicata in eliminazione di un elemento di tipo internalOrganizationUnit dell'oggetto radice.

Per avere maggiori dettagli sui possibili elementi fare riferimento alla definizione del modello, disponibile nella sezione [Modello Dati](#).

Infine è possibile applicare le validazioni, condizionalmente al soddisfacimento di determinate condizioni (opzionali). Queste condizioni sono specificate nella colonna "Applicabilità": se è specificato **always**, la validazione è sempre attiva.

Stato	Attori	MacroTipo	Tipo	Attributo/Identificativo	Applicabilita'
Bozza (draft)	all	enter	required	wfItemTypeId	always
				stringMap[applicationNumber]	always
				startPatentValidator	always
			complex	intellectualPropertyOwnerValidator	always
				checkCreationPermissionsValidator	always
				uniquePatentValidator	always
		save	required	link:priorityExtensionLink	isNotPriority
			complex	warningAcknowledgementValidator	isNotPriority
				bookEntryTotalPatentValidator	always
		delete	complex	lastLegalStatusCalculatorPatent	always
Validato (validated)	all	enter	required	priorityExtensionLinkDeleteValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
			complex	owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
			complex	owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
		save	complex	owner:save addOrgUnitFromNewOwnerValidator	always
		delete	complex	owner:save addOrgUnitFromNewOwnerValidator	always
		element	complex	owner:save addOrgUnitFromNewOwnerValidator	always
Archiviato (archived)	all	enter	required	owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
			complex	owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
			complex	owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
				owner:save addOrgUnitFromNewOwnerValidator	always
		save	complex	owner:save addOrgUnitFromNewOwnerValidator	always
		delete	complex	owner:save addOrgUnitFromNewOwnerValidator	always
		element	complex	owner:save addOrgUnitFromNewOwnerValidator	always

		delete	complex	priorityExtensionLinkDeleteValidator	always
				owner.save addOrgUnitFromNewOwnerValidator	always
		element	complex	priorityExtensionLink:delete notAllowedValidator	always
				priorityExtensionLink:save notAllowedValidator	always

Logiche (action/start)

Nella seguente sezione vengono riportate le

- **START LOGICS**
Le start logics sono le "azioni" che vengono eseguite in fase di creazione di un nuovo oggetto radice
- **ACTION LOGICS**
Le action logics sono delle "azioni" che vengono eseguite al verificarsi di determinati eventi.
Gli eventi contemplati sono:
 - enter: ingresso in uno stato
 - save: salvataggio dell'oggetto radice

Di seguito viene riportato il dettaglio delle logiche definite per questo flusso.

START LOGICS

- [wfStartLogicIdentifier](#)
- [wfStartLogicMultipleOwners](#)
- [wfStartLogicPatent](#)

ACTION LOGICS

- Bozza (draft)
 - **ENTER LOGICS**
 - [wfActionLogicEnterMailSender](#)
 - [wfActionLogicTransitionLogger](#)
 - **SAVE LOGICS**
 - [wfActionLogicSavePersonalVision - Responsabile \(owner\)](#)
 - [wfActionLogicSaveContextVision](#)
 - [wfActionLogicSaveNewYear](#)
- Validato (validated)
 - **ENTER LOGICS**
 - [wfActionLogicTransitionLogger](#)
 - [wfActionLogicEnterMailSender](#)
 - **SAVE LOGICS**
 - [wfActionLogicSavePersonalVision - Responsabile \(owner\)](#)
 - [wfActionLogicSaveContextVision](#)
 - [wfActionLogicSaveNewYear](#)
- Archiviato (archived)
 - **ENTER LOGICS**
 - [wfActionLogicEnterMailSender](#)
 - [wfActionLogicTransitionLogger](#)
 - **SAVE LOGICS**
 - [wfActionLogicSavePersonalVision - Responsabile \(owner\)](#)
 - [wfActionLogicSaveContextVision](#)
 - [wfActionLogicSaveNewYear](#)