

Scenari Layer Utente

Sigla	SV.0026.017
Versione Geoweb	4.8.3
Versione MapGuide	3.1.2
Data rilascio	2026-06-26
Redattore	GSC
Cliente	FIBERCOP ZENITH REAL ESTATE
Issue	#1809

1. Introduzione

Il presente documento descrive il rilascio della funzionalità **Scenari Layer Utente** (sigla SV.0026.017) per il widget **Layout 2D** del prodotto Geoweb.

Il documento è rivolto al team di implementazione e ai responsabili tecnici del progetto; ha lo scopo di descrivere in modo esaustivo le modifiche apportate al framework, le dipendenze tecniche e le istruzioni di installazione.

Documento di riferimento	Identificativo
Studio di Fattibilità / BRD	SV.0026.017.SF — Layout 2D Scenari Layer Utente — V02 — 30/03/2026
Progetto Esecutivo / PE	SV.0026.017.PE — Layout 2D Scenari Layer Utente — V02 — 12/05/2026
Cliente / Committente	FIBERCOP ZENITH REAL ESTATE
Sviluppo correlato	SV.0026.016 — Layer Utente WMS e Shapefile (prerequisito funzionale)

2. Contesto

2.1 Scenario precedente

FiberCop utilizza Geoweb per l'analisi territoriale effettuata da un gruppo ristretto di utenti (4-5 persone). Le analisi si basano sulla sovrapposizione di più layer cartografici, alcuni dei quali non sono pre-caricabili in quanto prodotti dinamicamente (es. simulazioni della protezione civile).

Nel framework Geoweb non esiste alcuno strumento gestibile dall'utente per la definizione di scenari di visualizzazione. I layer della mappa sono definiti staticamente tramite strumenti esterni; non esiste alcun meccanismo per salvare e richiamare rapidamente una specifica combinazione di layer accesi/spenti.

2.2 Scenario Attuale

Con questo rilascio l'utente finale può definire, salvare e richiamare **scenari di layer**: una combinazione nominata di layer della mappa in cui ciascun layer è impostato come acceso o spento per rappresentare una determinata situazione o punto di vista.

Gli scenari sono accessibili direttamente dalla scheda mappa 2D tramite una nuova toolbar integrata nel tab legenda. Gli scenari sono condivisi tra tutti gli utenti con accesso alla mappa. I permessi di creazione, modifica ed eliminazione sono governati dal meccanismo DACL di Geoweb.

La funzionalità è complementare a **SV.0026.016** (Layer Utente WMS e Shapefile): gli scenari fotografano lo stato di **tutti** i layer in legenda — layer di prodotto, layer dinamici, layer WMS utente e layer Shapefile utente — indipendentemente dalla loro origine.

3. Nomenclatura

Termine	Definizione
Scenario	Combinazione nominata di layer della mappa, in cui ciascun layer è impostato come acceso o spento. Fotografa lo stato della legenda in un momento specifico e permette di ripristinarlo.
Toolbar Scenari	Nuova toolbar introdotta nel tab legenda del widget Layout 2D; contiene select scenario, pulsanti Nuovo, Salva, Edita Nome, Elimina.
legend_state_json	Campo JSONB/CLOB della tabella gwd_mg_scenario che memorizza lo snapshot dello stato della legenda (id layer + flag visibilità).
legend_state_html	Campo HTML readonly generato dal trigger Groovy di classe, che rappresenta in forma leggibile il contenuto dello scenario.
DACL	Meccanismo Geoweb di controllo degli accessi a livello di classe: governa permessi di inserimento, modifica ed eliminazione degli scenari per utente/gruppo.
Snapshot legenda	Fotografia dello stato corrente dell'albero della legenda (layer accesi/spenti) eseguita nel momento in cui si crea o aggiorna uno scenario.

4. Ambito e Dipendenze

Componente	Versione / Dettaglio
Geoweb Framework	4.8.3
MapGuide Open Source	3.1.2
Database supportati	PostgreSQL, Oracle, SQL Server
Dipendenza funzionale	SV.0026.016 — Layer Utente WMS e Shapefile
Groovy trigger classe	Genera legend_state_html a partire da legend_state_json

5. Classe Geoweb

È stata introdotta una nuova classe Geoweb per la gestione degli scenari:

Classe Geoweb	Tabella DB	Label	Trigger
gwd_mg_scenario	gwd_mg_scenario	Scenari Layer Utente	Groovy trigger di classe: genera legend_state_html da legend_state_json

La gestione della classe avviene nella funzione applicativa **“Gestione Layer Utente e Scenari”** (evoluzione della funzione SV.0026.016), con accesso a lista e dettaglio standard. I metadati della classe si trovano in DR/metadati/ e vanno importati tramite il Webadmin Geoweb.

6. Schema Dati

6.1 Tabella gwd_mg_scenario

Campo	Tipo	Descrizione
id_gwd_mg_scenario	INTEGER PK	Identificativo univoco dello scenario (autoincrement)
gw_map_name	VARCHAR	Nome della mappa Geoweb associata allo scenario (NOT NULL)
scenario_name	VARCHAR	Nome dello scenario — univoco (UNIQUE, NOT NULL)
scenario_descr	VARCHAR	Descrizione testuale dello scenario (opzionale)
legend_state_json	JSONB/CLOB	Snapshot dello stato della legenda: array di oggetti {layerId, visible}
legend_state_html	VARCHAR/CLOB	Rappresentazione HTML readonly generata dal trigger Groovy
created_at	TIMESTAMP	Data e ora di creazione dello scenario
created_by	VARCHAR	Utente che ha creato lo scenario
updated_at	TIMESTAMP	Data e ora dell'ultimo aggiornamento
updated_by	VARCHAR	Utente che ha effettuato l'ultimo aggiornamento

7. Componenti Sviluppati

Componente	File	Descrizione
Classe Geoweb	gwd_mg_scenario (metadati)	Classe di prodotto con lista + dettaglio per gestione scenari
Groovy trigger	trigger di classe	Genera legend_state_html readable da legend_state_json
Toolbar scenari	LayersManagement.js	Nuova toolbar nel tab legenda: select, Nuovo, Salva, Edita Nome, Elimina
Controller	PluginLayersController.java	Operazioni backend per scenari (recupero, creazione, aggiornamento, eliminazione)
View	layers.jsp	Integrazione toolbar scenari nel tab legenda del widget Layout 2D

8. Funzionalità UI — Gestione dalla lista e dettaglio di classe

Attore: Utente Finale

8.1 Lista scenari

La lista mostra tutti gli scenari configurati per tutte le mappe, con i campi anagrafici (nome, mappa, date).

Scenari Layer Utente - (Nuovo)

Dati Preview

Scenario Name *

Descrizione

Sarà visualizzato lo Scenario Layer Utente

Creato da tester

Creato in data

Modificato da

Modificato in data

Chiudi Crea Crea e Chiudi

Fig. 2 — Dettaglio classe: nuovo scenario (vuoto)

8.3 Scenario esistente — dettaglio

Il dettaglio mostra i campi anagrafici dello scenario (nome, mappa, descrizione, date). Il campo `legend_state_html` mostra in forma readonly la lista dei layer inclusi nello scenario con il relativo stato acceso/spento.

The screenshot shows a web application window titled "Scenari Layer Utente: riserve+rischio_alluvione". It features a "Dati" tab and a "Preview" tab. The "Dati" tab contains the following fields:

Scenario Name *	riserve+rischio_alluvione	
Descrizione		
Mappa *	mapCartographic	
Creato da		
Creato in data	09-06-2026	00:47
Modificato da	tester	
Modificato in data	09-06-2026	11:39

At the bottom of the form, there are three buttons: "Chiudi", "Salva", and "Salva e Chiudi".

Fig. 3 — Dettaglio scenario esistente: campi anagrafici

8.4 Scenario esistente — preview HTML

Il widget HTML readonly generato dal trigger Groovy mostra in forma leggibile il contenuto dello snapshot della legenda.

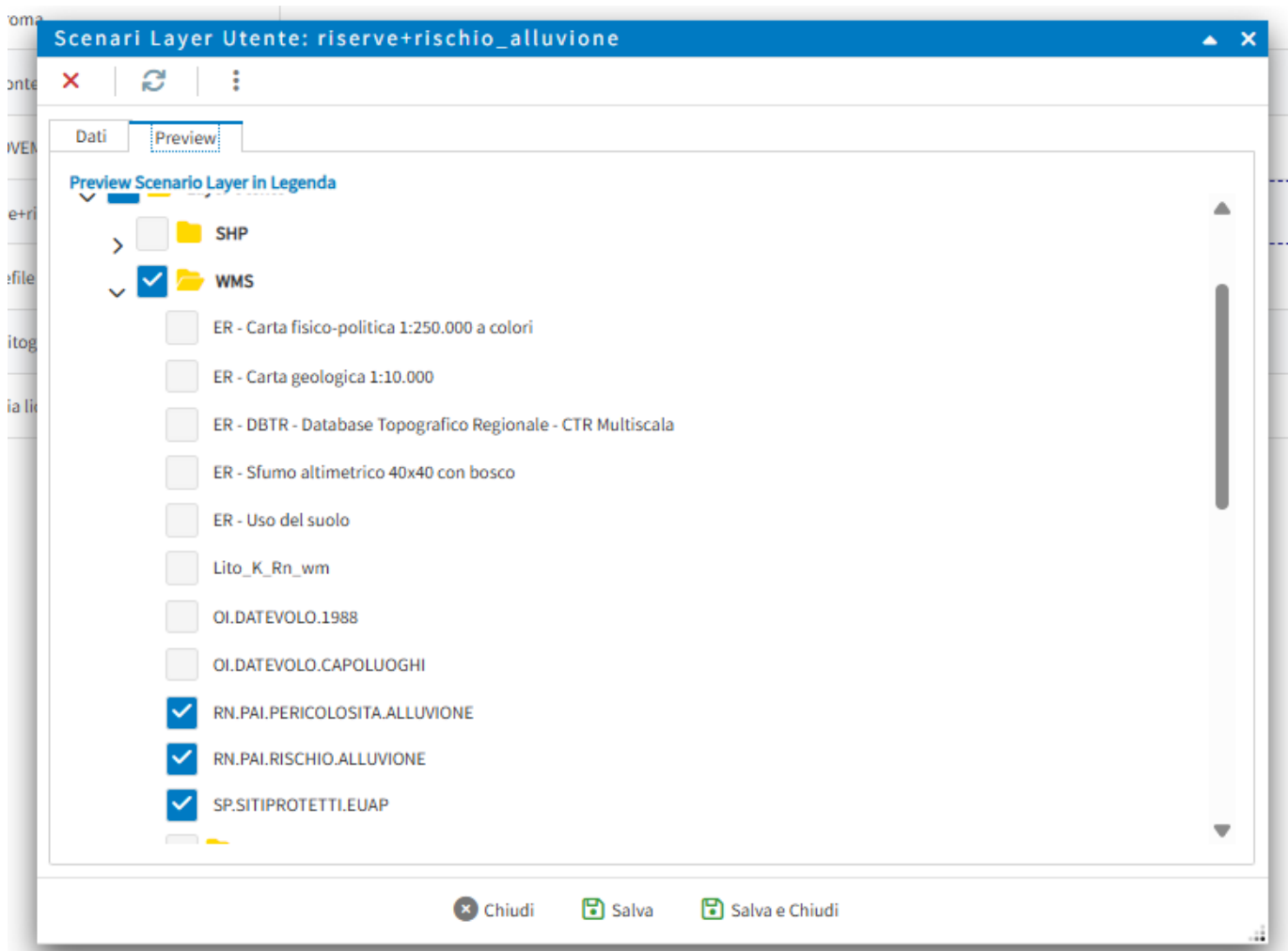


Fig. 4 — Dettaglio scenario esistente: preview HTML del legend_state

9. Funzionalità UI — Toolbar Scenari nella Mappa

Attore: Utente Finale

9.1 Toolbar nel tab legenda

La toolbar scenari è integrata nel tab legenda del widget Layout 2D. Si inizializza solo dopo che tutti i layer (di prodotto, dinamici, WMS utente e Shapefile utente) sono stati caricati completamente.

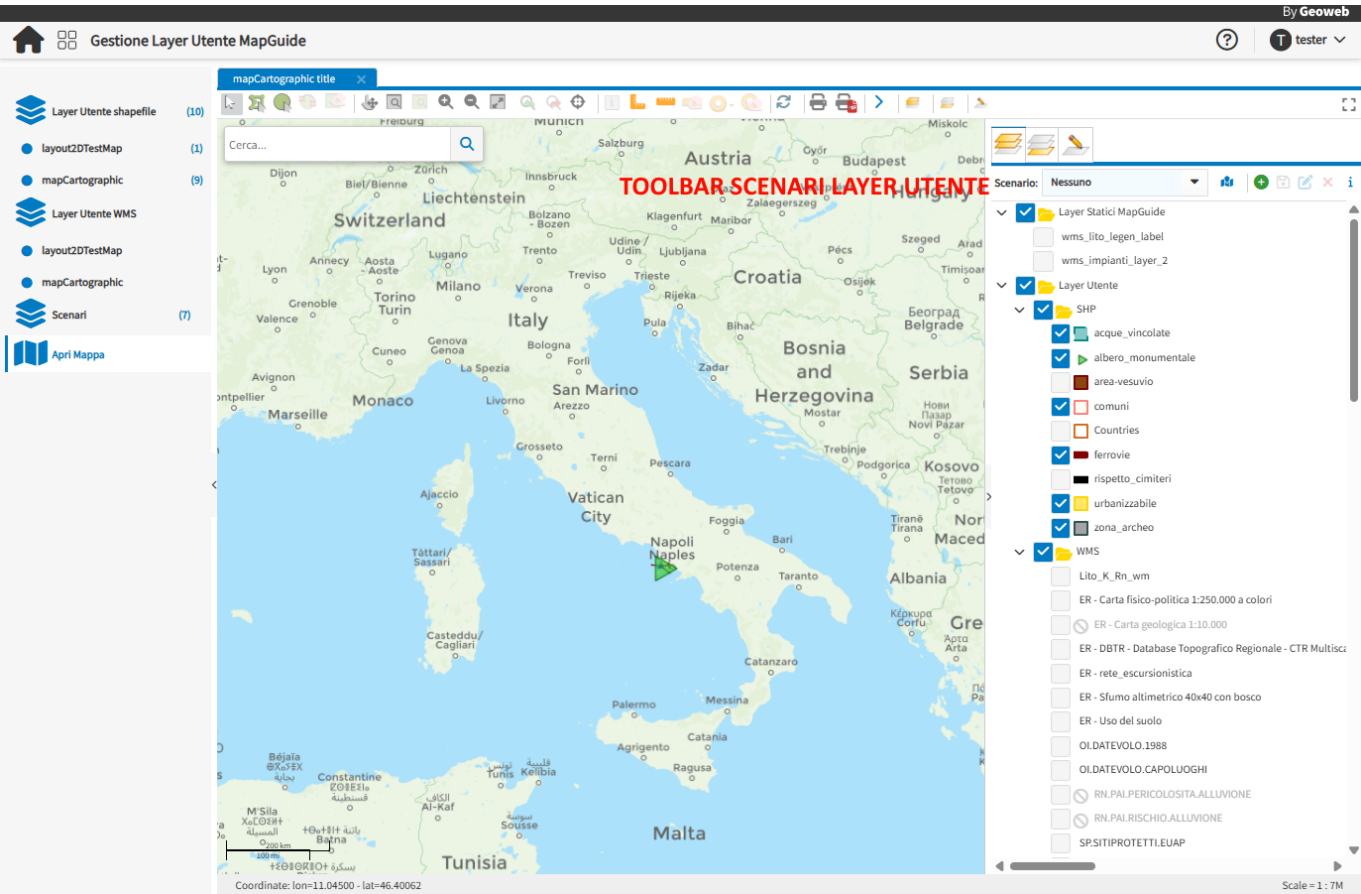


Fig. 5 — Toolbar scenari nel tab legenda del widget Layout 2D

9.2 Dettaglio pulsanti toolbar

La toolbar contiene: select scenario (filtra per mappa corrente, include “Nessuno scenario impostato”), pulsante **Nuovo**, pulsante **Salva**, pulsante **Edita Nome**, pulsante **Elimina**.



Fig. 6 — Dettaglio pulsanti della toolbar scenari

Elemento	Visibilità	Comportamento
Select scenario	Sempre visibile	Applica lo scenario selezionato alla legenda; “Nessuno scenario impostato” = nessuna applicazione
Pulsante Nuovo	Se l'utente ha permesso di creazione (DACL)	Apre input nome; al conferma salva snapshot della legenda come nuovo scenario
Pulsante Salva	Se scenario selezionato + utente ha permesso modifica + legenda modificata	Sovrascrive legend_state_j son con stato corrente
Pulsante Edita Nome	Se scenario selezionato + utente ha permesso modifica	Apre input con nome corrente editabile
Pulsante Elimina	Se scenario selezionato + utente ha permesso eliminazione	Richiede conferma; elimina scenario; reimposta select

9.3 Creazione nuovo scenario da mappa

L'Utente Finale preme **Nuovo**: appare un input per il nome dello scenario. Alla conferma, il sistema fotografa lo stato corrente della legenda e crea il nuovo scenario.

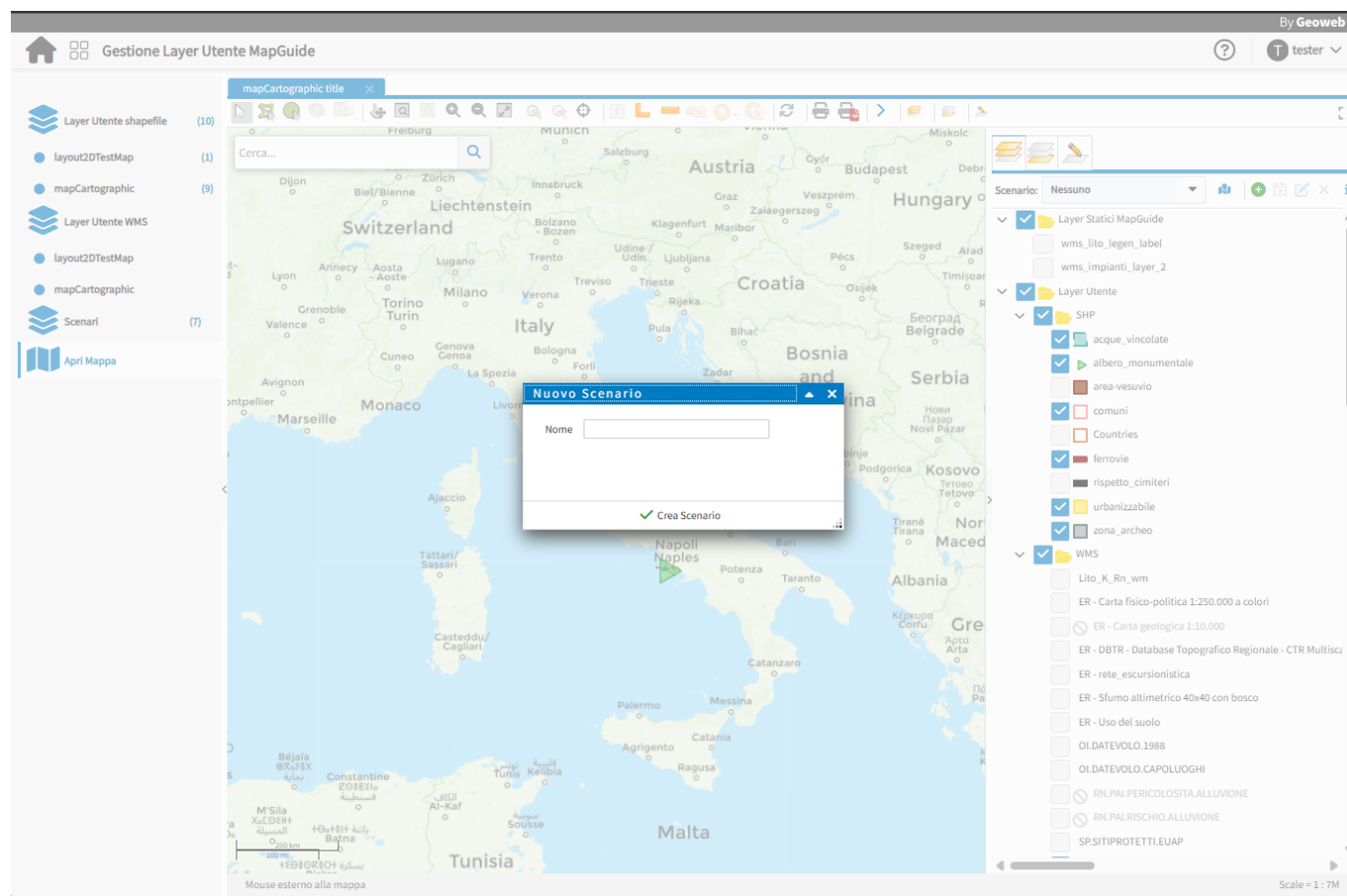


Fig. 7 — Dialogo creazione nuovo scenario da mappa: input nome

9.4 Scenario applicato in mappa

Dopo la selezione dalla select, la legenda si aggiorna accendendo e spegnendo i layer secondo lo snapshot salvato. Il nome dello scenario corrente è evidenziato nella toolbar.

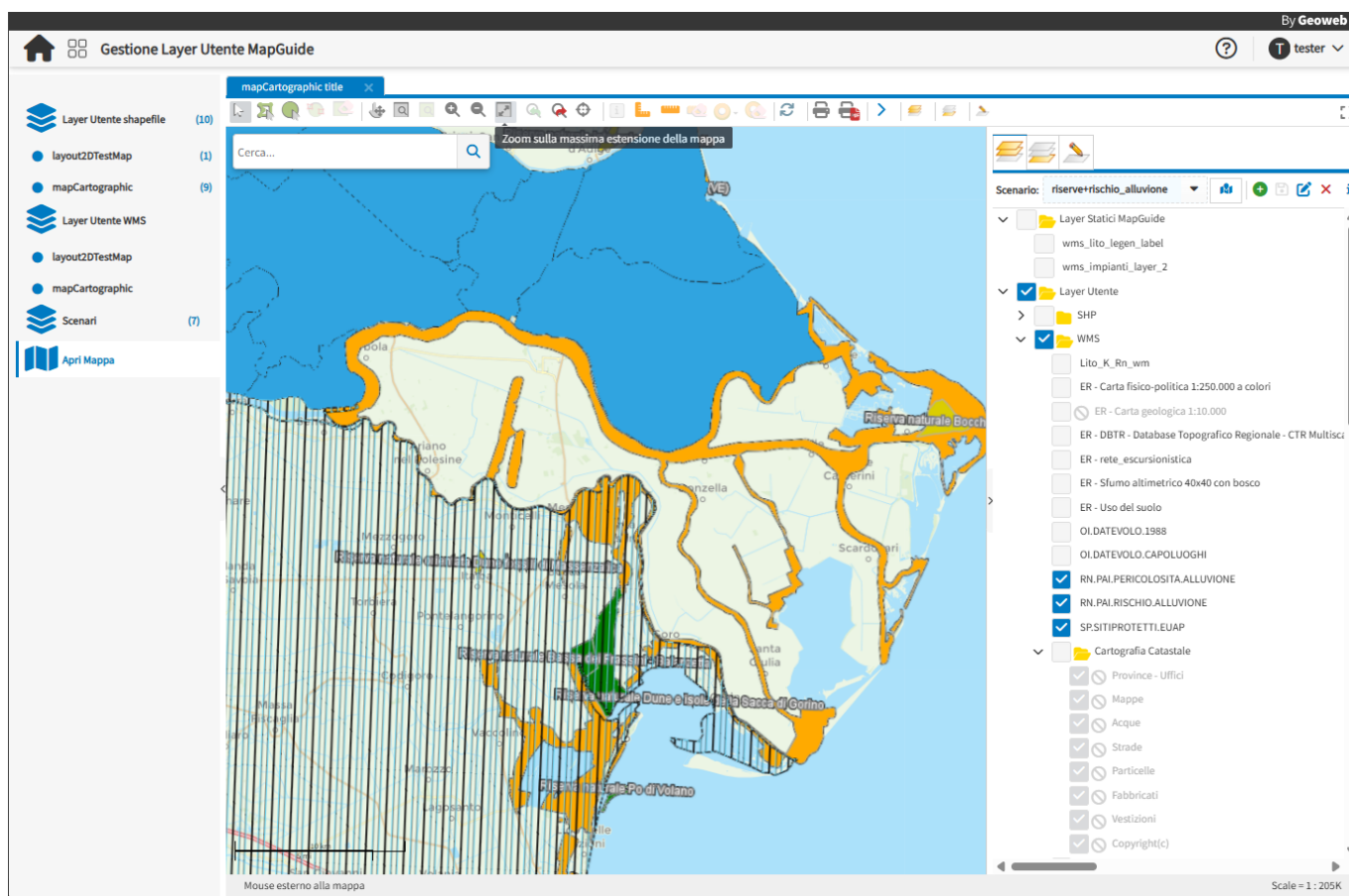


Fig. 8 — Scenario applicato: legenda aggiornata coerentemente

9.5 Modifica nome scenario

L'Utente Finale preme **Edita Nome**: appare un input con il nome corrente, editabile. Alla conferma il nome viene aggiornato senza modificare il `legend_state_j`son.

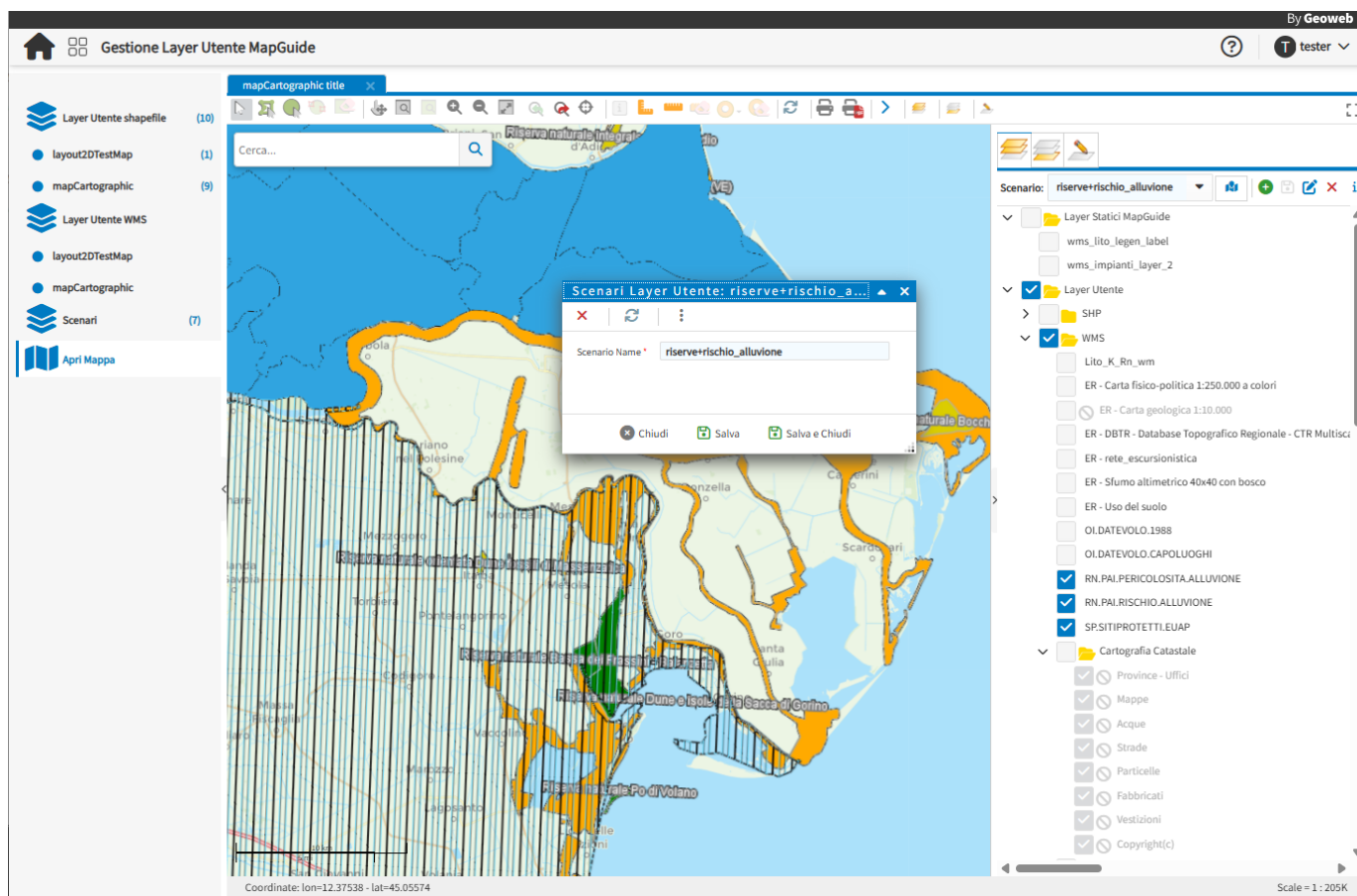


Fig. 9 — Dialogo modifica nome scenario

9.6 Conferma eliminazione scenario

L'Utente Finale preme **Elimina**: il sistema chiede conferma esplicita. Alla conferma, lo scenario viene eliminato; la mappa mantiene lo stato corrente; la select torna a "Nessuno scenario impostato".

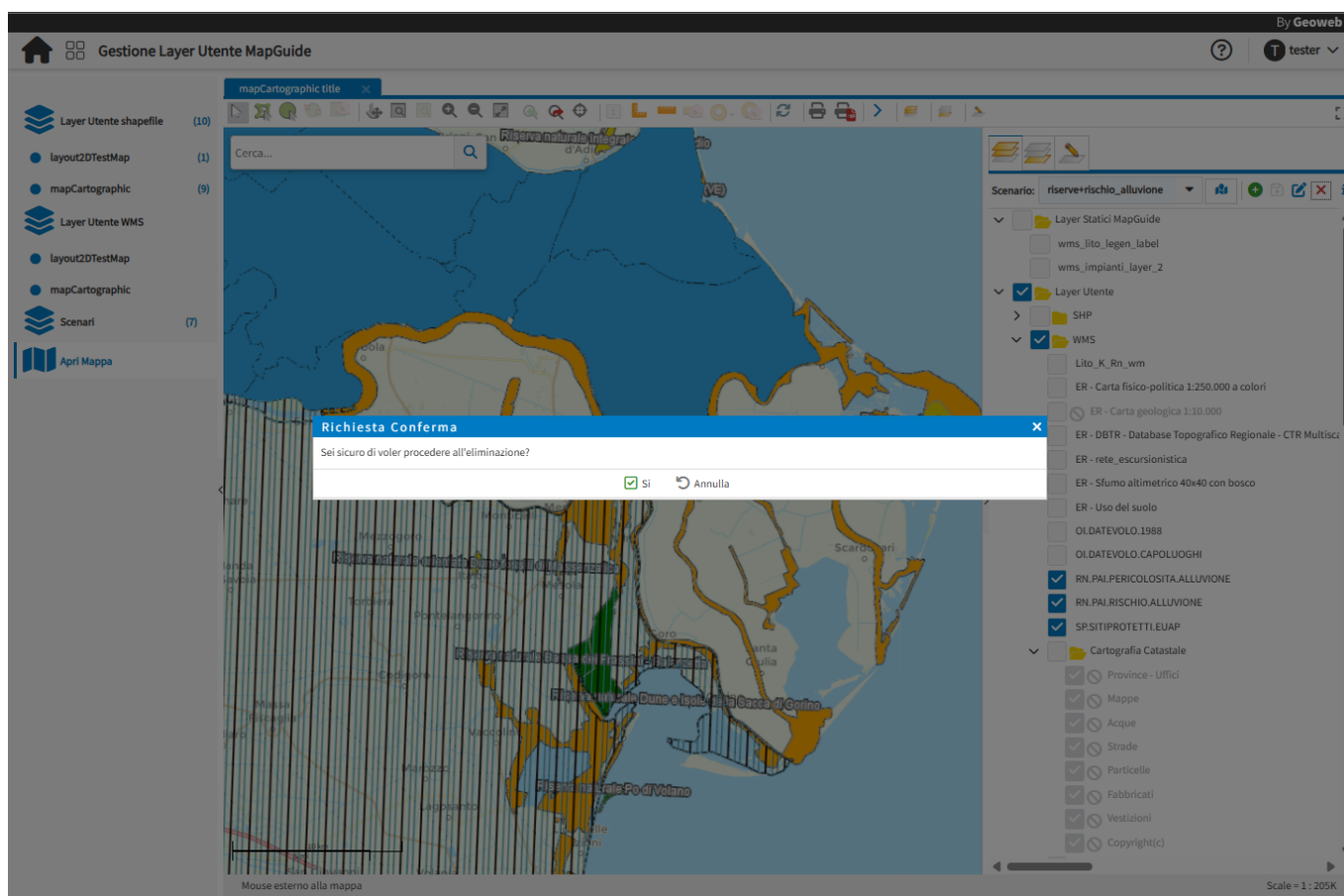


Fig. 10 — Dialogo conferma eliminazione scenario

10. Note Tecniche

10.1 Sequenza di inizializzazione

La toolbar scenari viene inizializzata **dopo** che tutti i layer della legenda sono pronti. La sequenza garantita è:

1. apertura mappa Layout 2D
2. caricamento layer di prodotto e dinamici
3. caricamento layer utente WMS e Shapefile (SV.0026.016)
4. costruzione albero della legenda completo
5. **inizializzazione toolbar scenari**
6. applicazione eventuale dello scenario precedentemente selezionato

Questa sequenza è vincolante: inizializzare gli scenari prima che la legenda sia completa produrrebbe snapshot incompleti o applicazioni parziali.

10.2 Gestione stato modificato non salvato

Il client confronta continuamente lo stato corrente della legenda con lo snapshot dello scenario

applicato (confronto su id layer + flag visibilità). Se rileva una differenza, attiva l'indicatore “modificato non salvato” rendendo attivo il pulsante **Salva** e disattivando il richiamo automatico.

10.3 Resilienza a layer mancanti

Se un layer presente nello snapshot non è disponibile nella legenda corrente (es. layer utente eliminato), il sistema **non blocca** l'applicazione dello scenario: continua su tutti gli altri layer presenti, ignorando silenziosamente quelli mancanti.

10.4 Operazioni backend

Le operazioni sugli scenari utilizzano le API JS standard di classe di Geoweb:

- `insertGwClassRecord()` — creazione scenario
- `updateGwClassRecord()` — aggiornamento stato / nome
- `deleteGwClassRecord()` — eliminazione scenario

Il backend filtra sempre per `gw_map_name`, garantendo che ogni mappa veda solo i propri scenari.

10.5 Groovy trigger — `legend_state_html`

Il trigger Groovy di classe genera il campo `legend_state_html` a partire da `legend_state_json`. Il widget HTML prodotto mostra in forma readonly la lista dei layer inclusi nello scenario con il relativo stato acceso/spento. Questo campo è di sola lettura nel dettaglio di classe: non è prevista né supportata la modifica manuale di `legend_state_json` dall'interfaccia.

10.6 Complementarità con SV.0026.016

Gli scenari non hanno conoscenza del tipo di layer; si limitano a fotografare e applicare la visibilità di ciò che la legenda espone. Il caricamento dei layer WMS utente e Shapefile utente è onere di SV.0026.016. La sequenza esecutiva garantisce che gli scenari vedano sempre la legenda completa, inclusi i layer utente.

11. Ambiente di Test

Voce	Valore
URL applicazione Utente Finale	http://192.168.0.225:8090/test_gw48/project/mg_user_layer.html?group=TESTER_GROUP
Utente Finale — username	tester

Voce	Valore
Utente Finale — password	Tester0@
URL Webadmin	http://192.168.0.225:8090/test_gw48admin/index.html
Webadmin — utente	adminTester
Webadmin — password	admin
Versione Geoweb	4.8.3
Versione MapGuide	3.1.2

12. Script SQL — Data Schema

Script DDL per la creazione della tabella applicativa gwd_mg_scenario. Eseguire prima di importare i metadati.

12.1 PostgreSQL — data_script.sql

[data_script_postgres.sql](#)

```
-- SV.0026.017 - Layout 2D: Scenari Layer Utente
-- PostgreSQL DDL
-- Redattore: GSC Data: 2026-06-26

-- Table: test_gw48data.gwd_mg_scenario

-- DROP TABLE IF EXISTS test_gw48data.gwd_mg_scenario;

CREATE TABLE IF NOT EXISTS test_gw48data.gwd_mg_scenario
(
    id_gwd_mg_scenario INTEGER NOT NULL,
    gw_map_name CHARACTER VARYING COLLATE pg_catalog."default" NOT
NULL,
    scenario_name CHARACTER VARYING COLLATE pg_catalog."default" NOT
NULL,
    scenario_descr CHARACTER VARYING COLLATE pg_catalog."default",
    legend_state_json jsonb,
    created_at TIMESTAMP WITHOUT TIME zone,
    created_by CHARACTER VARYING COLLATE pg_catalog."default",
    updated_at TIMESTAMP WITHOUT TIME zone,
    updated_by CHARACTER VARYING COLLATE pg_catalog."default",
    legend_state_html CHARACTER VARYING COLLATE pg_catalog."default",
    CONSTRAINT gwd_mg_scenario_pkey PRIMARY KEY (id_gwd_mg_scenario),
```

```

        CONSTRAINT unique_scenario_name UNIQUE (scenario_name)
    )
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;

ALTER TABLE IF EXISTS test_gw48data.gwd_mg_scenario
    OWNER TO test_gw48data;

-- =====
-- Permessi: schema dati -> schema metadati
-- Il widget della classe gwd_mg_scenario legge gwm_maps
-- nello schema metadati per popolare la select mappa.
-- Adattare i nomi schema all'ambiente di destinazione.
-- =====
GRANT SELECT ON test_gw48metadata.gwm_maps TO test_gw48data;

```

12.2 Oracle — data_script.sql

[data_script_oracle.sql](#)

```

-- SV.0026.017 - Layout 2D: Scenari Layer Utente
-- Oracle DDL
-- Redattore: GSC Data: 2026-06-26

-- =====
-- Tabella: GWD_MG_SCENARIO
-- =====

CREATE TABLE GWD_MG_SCENARIO
(
    ID_GWD_MG_SCENARIO    NUMBER(10)    NOT NULL,
    GW_MAP_NAME           VARCHAR2(500)  NOT NULL,
    SCENARIO_NAME         VARCHAR2(500)  NOT NULL,
    SCENARIO_DESCR        VARCHAR2(2000),
    LEGEND_STATE_JSON     CLOB,
    LEGEND_STATE_HTML     CLOB,
    CREATED_AT            TIMESTAMP,
    CREATED_BY            VARCHAR2(200),
    UPDATED_AT            TIMESTAMP,
    UPDATED_BY            VARCHAR2(200),
    CONSTRAINT GWD_MG_SCENARIO_PK        PRIMARY KEY
    (ID_GWD_MG_SCENARIO),
    CONSTRAINT UNIQUE_SCENARIO_NAME      UNIQUE (SCENARIO_NAME)
);

-- =====

```

```
-- Permessi: utente dati -> utente metadati (GWM_MAPS)
-- Il widget della classe GWD_MG_SCENARIO legge GWM_MAPS
-- nello schema metadati per popolare la select mappa.
-- Eseguire connessi come utente metadati o come DBA.
-- Adattare GW48DATA all'utente dati dell'ambiente.
-- =====

-- Eseguire come utente dello schema metadati:
GRANT SELECT ON GWM_MAPS TO GW48DATA;

-- oppure da DBA (adattare entrambi i nomi utente):
-- GRANT SELECT ON GW48METADATA.GWM_MAPS TO GW48DATA;
```

12.3 SQL Server — data_script.sql

[data_script_mssqlserver.sql](#)

```
-- SV.0026.017 - Layout 2D: Scenari Layer Utente
-- SQL Server DDL
-- Redattore: GSC Data: 2026-06-26

-- =====
-- Tabella: GWD_MG_SCENARIO
-- =====

IF OBJECT_ID(N'dbo.GWD_MG_SCENARIO', N'U') IS NULL
BEGIN
    CREATE TABLE dbo.GWD_MG_SCENARIO
    (
        ID_GWD_MG_SCENARIO    INT                NOT NULL,
        GW_MAP_NAME           NVARCHAR(500)       NOT NULL,
        SCENARIO_NAME         NVARCHAR(500)       NOT NULL,
        SCENARIO_DESCR        NVARCHAR(2000),
        LEGEND_STATE_JSON     NVARCHAR(MAX),
        LEGEND_STATE_HTML     NVARCHAR(MAX),
        CREATED_AT            DATETIME2,
        CREATED_BY            NVARCHAR(200),
        UPDATED_AT            DATETIME2,
        UPDATED_BY            NVARCHAR(200),
        CONSTRAINT GWD_MG_SCENARIO_PK    PRIMARY KEY
        (ID_GWD_MG_SCENARIO),
        CONSTRAINT UNIQUE_SCENARIO_NAME    UNIQUE (SCENARIO_NAME)
    );
END
GO

-- =====
```

```
-- Permessi: utente dati -> schema metadati (gwm_maps)
-- Il widget della classe GWD_MG_SCENARIO legge gwm_maps
-- nello schema metadati per popolare la select mappa.
-- Adattare [<schema_metadati>] e [test_gw48data] all'ambiente.
-- =====
GRANT SELECT ON [<schema_metadati>].[gwm_maps] TO [test_gw48data];
GO
```

13. Installazione

Eseguire i passi nell'ordine indicato:

1. Eseguire DR/script/<dbms>/data_script.sql — crea la tabella gwd_mg_scenario e concede il permesso SELECT su gwm_maps allo schema dati
2. Importare i file XML da DR/metadata/ tramite il Webadmin Geoweb (Import/Export metadati):
 - gw_exp.classes gwd_mg_scenario.xml
 - gw_exp.projects mg_user_layer.xml
 - gw_exp.themes MAPGUIDE USER LAYER E SCENARIO.xml
3. Verificare che il trigger Groovy di classe sia configurato e generi correttamente legend_state_html
4. **Prerequisito:** SV.0026.016 deve essere già installato nell'ambiente

From:
<https://wiki.geowebframework.com/> - GeowebFramework

Permanent link:
https://wiki.geowebframework.com/doku.php?id=gwusermanual:interface:interface:scenari_layer:scenario_layer

Last update: 2026/06/29 10:06

