



una scelta comune

**LIFE09 ENV/IT/000056**



Il progetto è  
realizzato con il  
contributo  
finanziario del  
Programma LIFE  
della Commissione  
Europea

**Acronimo del progetto**  
*Project Acronym* **W.I.Z.**

**Titolo completo del  
progetto**  
*Project Full Title* **WIZ – WaterIZE spatial  
planning: encompass  
future drinkwater  
management conditions  
to adapt to climate  
change**

**Numero del progetto**  
*Project No* **LIFE09 ENV/IT/000056**

**No. Deliverable** **D11.2**

**Motore WIZ**

**WIZ Engine**

**Mese/Month – Anno/Year** **Giugno/June 2012**

## Partner di progetto/Project Partner



### Capofila/Main Contractor

**Acque S.p.A.**  
**Sede Legale: Via Garigliano, 1**  
**I - 50053 EMPOLI -IT**  
**Sede operativa: Via A. Bellatalla, 1**  
**I - 56121 Ospedaletto (PI)**  
<http://www.acque.net>

**Autorità di bacino**  
**Via dei Servi, 15**  
**I - 50122 FIRENZE - IT**  
<http://www.adiba.it>

**Ingegnerie Toscane S.r.l.**  
**Via di Villamagna, 90**  
**I - 50126 Firenze**

**Via A. Bellatalla, 1**  
**I - 56121 Ospedaletto (PI)**  
<http://www.acque.net>

**Instituto Tecnológico de Galicia**  
**PO.CO.MA.CO Sector I Portal 5**  
**ES - 15190 A Coruña - Galicia -**  
**ESPAÑA**  
<http://www.itg.es>

## Informazioni sul documento / Document Information

### Project / Progetto

Acronimo del progetto/  
Project Acronym

W.I.Z.

Titolo completo del progetto / Project Full  
Title

WIZ – WaterIZe spatial  
planning: encompass  
future drinkwater  
management conditions  
to adapt to climate  
change

Data di avvio / Project start:

09/01/10

Durata del Progetto / Project duration:

36 mesi

Contratto no / Grant agreement no.:

LIFE09 ENV/IT/000056

### Document

No Deliverable / Deliverable No:

D11.2

Titolo del Deliverable / Deliverable title:

Motore WIZ / WIZ Engine

Data contrattuale del

29/06/12

Deliverable / Contractual Date of Delivery:

Data di consegna del Deliverable / Actual

29/06/12

Date of Delivery:

Editore(i) / Editor(s):

Autore(i) / Author(s):

Revisore(i) / Reviewer(s):

Partner / Partner(s):

INGTOS

No Work package / Work package no.:

AZIONE #011

Titolo Work package / Work package title:

Implementazione del  
motore di proiezione WIZ

Leader del Work package / Work package  
leader:

INGTOS

Distribuzione / Distribution

Public

(Public / Reserved):

Natura / Nature (Report, ...):

Report

Versione-Revisione / Version-Revision:

0b

Bozza-Definitivo / Draft-Final

Final

No di pagine (inclusa copertina) / Total

54

number of pages:

(including cover)

Parole chiave / Keywords:

W.I.Z., Deliverable

## Revisioni/Change Log

<b>Motivo della revisione/Reason for change</b>	<b>Argomento della revisione/Issue</b>	<b>Numero della Revisione/Revi sion</b>	<b>Data della Revisione/D ate</b>
--	Versione Iniziale/Initial Version	0a	29/06/12
<b>Revisione e Aggiornamenti/U pdated Executive Summary in EN</b>	Inserimento sintesi EN/Updated Executive Summary in EN	0b	04/02/201 3



## **Esonero Responsabilità/Disclaimer**

Questo documento contiene descrizioni che riguardano le attività, i risultati e i prodotti del Progetto WIZ. Alcune sue parti potrebbero essere tutelate sotto Diritto di Proprietà Intellettuale (IPR). Per questo motivo vi chiediamo di contattare il Consorzio WIZ prima di utilizzarlo (e.mail: o.cei@ingegnerietoscane.net).

Se ritenete che questo documento sia in qualsiasi modo lesivo dei diritti di proprietà intellettuale di vostro possesso – come persona o come rappresentante di un organizzazione – informateci tempestivamente. Gli autori di questo documento hanno preso tutte le misure disponibili possibili per far sì che il suo contenuto sia accurato, consistente e legale. Tuttavia, né il partenariato nel suo insieme, né i singoli partner che direttamente o indirettamente abbiano preso parte alla creazione e alla pubblicazione di questo documento sono responsabili per qualsiasi cosa possa accadere come risultato del suo utilizzo.

Questa pubblicazione è stata realizzata grazie al contributo dell'Unione Europea. Il consorzio WIZ è il solo responsabile del contenuto di questa pubblicazione che non riflette necessariamente il pensiero dell'Unione Europea

WIZ è parzialmente finanziato dall'Unione Europea (Life+ Programme).

*This document contains description of the WIZ project findings, work and products. Certain parts of it might be under partner Intellectual Property Right (IPR) rules so, prior to using its content please contact the consortium head for (e.mail: o.cei@acqueingegneria.net).*

*In case you believe that this document harms in any way IPR held by you as a person or as a representative of an entity, please do notify us immediately.*

*The authors of this document have taken any available measure in order for its content to be accurate, consistent and lawful. However, neither the project consortium as a whole nor the individual partners that implicitly or explicitly participated the creation and publication of this document hold any sort of responsibility that might occur as a result of using its content.*

*This publication has been produced with the assistance of the European Union. The content of this publication is the sole responsibility of WIZ Consortium and can in no way be taken to reflect the views of the European Union.*

*WIZ is a project partially funded by the European Union*

## Indice

<a href="#">W.I.Z.....</a>	<a href="#">1</a>
<a href="#">WIZ – WaterIZe spatial planning: encompass future drinkwater management conditions to adapt to climate change .....</a>	<a href="#">1</a>
<a href="#">LIFE09 ENV/IT/000056.....</a>	<a href="#">1</a>
<a href="#">Motore WIZ.....</a>	<a href="#">1</a>
<a href="#">WIZ Engine.....</a>	<a href="#">1</a>
<a href="#">Partner di progetto/Project Partner.....</a>	<a href="#">2</a>
<a href="#">Informazioni sul documento/Document Information.....</a>	<a href="#">3</a>
<a href="#">    Project/Progetto.....</a>	<a href="#">3</a>
<a href="#">    Document.....</a>	<a href="#">3</a>
<a href="#">Revisioni/Change Log.....</a>	<a href="#">3</a>
<a href="#">Esonero Reponsabilità/Disclaimer.....</a>	<a href="#">5</a>
<a href="#">Indice.....</a>	<a href="#">6</a>
<a href="#">1 Executive summary.....</a>	<a href="#">9</a>
<a href="#">2 Sommario esecutivo.....</a>	<a href="#">12</a>
<a href="#">3 Scopo.....</a>	<a href="#">13</a>
<a href="#">4 Framework.....</a>	<a href="#">13</a>
<a href="#">5 Il patrimonio informativo.....</a>	<a href="#">14</a>
<a href="#">6 L'ARCHITETTURA.....</a>	<a href="#">16</a>
<a href="#">    6.1 Il Database.....</a>	<a href="#">18</a>
<a href="#">        6.1.1 notifications.....</a>	<a href="#">20</a>
<a href="#">            6.1.1.1 notification categories.....</a>	<a href="#">20</a>
<a href="#">            6.1.1.2 settings.....</a>	<a href="#">21</a>
<a href="#">            6.1.1.3 users.....</a>	<a href="#">21</a>
<a href="#">            6.1.1.4 roles.....</a>	<a href="#">21</a>
<a href="#">            6.1.1.5 water qualities.....</a>	<a href="#">22</a>
<a href="#">            6.1.1.6 water quality options.....</a>	<a href="#">22</a>
<a href="#">            6.1.1.7 water faults.....</a>	<a href="#">22</a>
<a href="#">            6.1.1.8 water fault options.....</a>	<a href="#">22</a>
<a href="#">            6.1.1.9 water requests.....</a>	<a href="#">22</a>
<a href="#">            6.1.1.10 water request history.....</a>	<a href="#">23</a>
<a href="#">            6.1.1.11 water request geometries.....</a>	<a href="#">23</a>
<a href="#">            6.1.1.12 water request geometry zones.....</a>	<a href="#">23</a>
<a href="#">            6.1.1.13 water request geometry zone properties.....</a>	<a href="#">23</a>
<a href="#">            6.1.1.14 water request parameters.....</a>	<a href="#">24</a>
<a href="#">            6.1.1.15 zones.....</a>	<a href="#">24</a>
<a href="#">            6.1.1.16 zones water request parameters.....</a>	<a href="#">24</a>
<a href="#">            6.1.1.17 water demand formulas.....</a>	<a href="#">24</a>
<a href="#">            6.1.1.18 water supply.....</a>	<a href="#">24</a>
<a href="#">            6.1.1.19 city state boundaries.....</a>	<a href="#">25</a>
<a href="#">            6.1.1.20 operative margin.....</a>	<a href="#">25</a>
<a href="#">            6.1.1.21 service area.....</a>	<a href="#">25</a>
<a href="#">            6.1.1.22 service area operative margin.....</a>	<a href="#">25</a>
<a href="#">            6.1.1.23 plugins.....</a>	<a href="#">25</a>
<a href="#">            6.1.1.24 sendmail logs.....</a>	<a href="#">26</a>

<a href="#">6.2 Controllo degli accessi.....</a>	<a href="#">26</a>
<a href="#">6.3 Le notifiche.....</a>	<a href="#">27</a>
<a href="#">6.4 I plugins.....</a>	<a href="#">28</a>
<a href="#">6.5 Gli utenti.....</a>	<a href="#">29</a>
<a href="#">6.6 Le richieste di risorsa idrica.....</a>	<a href="#">30</a>
<a href="#">6.7 Interfaccia EPANET.....</a>	<a href="#">39</a>
<a href="#">6.8 Qualità Partecipata.....</a>	<a href="#">42</a>
<a href="#">6.9 Le informazioni geografiche.....</a>	<a href="#">43</a>
<a href="#">6.10 Log &amp; Feedback.....</a>	<a href="#">44</a>
<a href="#">6.11 Internazionalizzazione.....</a>	<a href="#">44</a>
<a href="#">6.12 Configurazione parametri.....</a>	<a href="#">45</a>
<a href="#">7 Glossario.....</a>	<a href="#">48</a>
<a href="#">8 Appendice A: Schema del Database.....</a>	<a href="#">49</a>

## **Lista delle immagini**

Illustration 1: MVC.....	10
Illustration 2: Applicazione Yii.....	11
Illustration 3: Architettura.....	13
Illustration 4: Workflow dell'applicazione.....	14
Illustration 5: Database.....	16
Illustration 6: Grafo controllo accessi.....	23
Illustration 7: Struttura delle notifiche.....	24
Illustration 8: Procedura di login.....	26
Illustration 9: Relazione tra water_requests, water_request_geometries e water_request_geometry_zones.....	27
Illustration 10: Procedura per il calcolo degli abitanti equivalenti e dell'idroesigenza.....	28
Illustration 11: Formule per il calcolo dell'idroesigenza.....	28
Illustration 12: Parametri, indici di conversione e loro relazione con le destinazioni d'uso.....	29
Illustration 13: Destinazioni d'uso organizzate gerarchicamente.....	30
Illustration 14: Procedura per il recupero dell'indice di conversione per il calcolo degli abitanti equivalenti.....	30
Illustration 15: Procedura per il recupero della formula per il calcolo dell'idroesigenza.....	31
Illustration 16: Valutazione degli indici di conversione.....	32
Illustration 17: Valutazione delle formule per il calcolo dell'idroesigenza .....	33
Illustration 18: Dettaglio calcolo idroesigenza.....	33
Illustration 19: Confronto tra idroesigenza e margine operativo.....	35
Illustration 20: Campi di una Junction, secondo la specifica EPANET...	36
Illustration 21: Struttura di un file di modello EPANET.....	37
Illustration 22: Generazione di un nuovo file EPANET.....	38
Illustration 23: Schema database.....	46

## 1 Executive summary

The purpose of deliverable **D11.2 – Wiz Engine** is to define the components of the core architecture of the WIZ platform, which is the framework required to connect to higher-level applications, *WIZ4Planner* and *WIZ4All*. In addition, the source code of the WIZ engine is unrestrictedly available and downloadable at <http://wiz.acque.net/index.php?r=site/page&view=source>.

The document is opened by a description of the supporting framework, around which the software has been organised and designed (*Basic Framework*). The WIZ portal is based on the Open Source framework **Yii** (<http://www.yiiframework.com/>), which has been written in PHP, one of the most commonly used web programming languages. Yii implements the **Model-View-Controller (MVC)** design pattern, where the login is separated from the user interface, so developers may easily change the login without affecting the interface.

As to the wealth of information that underpins the system's development, it will include additional information, both geographic and non geographic, some of which will be physically built into the system, while other information will be used on demand when accessing the W\*S services, according to the OGC standard. Enquiries are usually about: administrative boundaries of the municipalities served by the provider; servicing areas a municipality is made up of; availability table of municipalities and servicing areas; operating margin, both current and potential, in one or more prospective scenarios; table of mean yearly consumption and highest daily consumption of the water resource; DEM land elevation model. Information from *Orthophotographs* (geo-referenced aerial photographs, downloaded on demand from the GEOscopio service of the Region Tuscany), *CTR* (regional technical map, downloaded on demand from the same service), *distribution network, sources* (streams, lakes, wells, springs), *plants* (water purification, water feeding pipes, storage, pumping), *street directory* (geo-referenced street-, square- and place-names) will be used as well.

Actually, the system has been designed to interact with a geographic database. Data are stored in a **Postgres database with the PostGIS** extension. The **GeoServer** map server is used to view geographic information and forward it to other parties, and then it is forwarded through an **OGC standard** using the **WFS/WMS services**. Such architecture is completed by a **set of Open Source Java Script libraries** through which the map server may be easily queried and features may be added to the user interface.

If we look into the details, we can see which parts such architecture is composed of:

- **Database:** WIZ uses a spatially extended database to store data. The database is composed of a number of tables, which are detailed in the text and summarised in the *Database pattern*, enclosed herewith as an Appendix;
- **Access control:** it is based on an authorisation, which may be: a *role*, *task* or *operation*. A role is composed of multiple tasks, each one consisting of multiple operations, and an operation is an atomic permit. To make the system more flexible, each unit (role, task or operation) may be associated with a *rule*;
- **Notices:** they are automatically generated by the system when a given event occurs. Each notice is classed into a category depending on the unit involved (users, water resource requests, ...) and the event that generated it (create, update, ...);
- **Plugins:** they are uploaded by the developer as archives and add up to the system's features. A plugin is just like an MVC component so it easily integrates with the system and remarkably streamlines the plugin installation procedure;
- **Users:** the user categories that may interact with the system are: non-logged users and logged users. A logged user may play the role of a Planner, Water resource provider (technical department), Water resource provider (administration department), water resource authority, citizen, System manager, developer, systems engineer. The system also interacts with a W\*S server to collect geographic information. A user who wants to use the system (with most features) must first register by entering his/her first name (compulsory), surname (compulsory), municipality (compulsory), organisation, position, email address (compulsory), username (compulsory), password (compulsory) and role (compulsory). Once registered, the user must be authenticated by logging in and entering his/her username (compulsory) and password (compulsory);
- **Water resource requests:** they are the most complex component of the system and it is because of them that a planner may create and submit a water resource request to the provider; for further information about a request procedure and status, see deliverable *D12.1 – Test cases for the implementation of WIZ services to planners*. A water resource request may be defined by the following information: calculation of population equivalent and water requirement (this is automatically calculated by the system with the help of the user, who must answer some simple questions), formulae (to calculate the water requirement associated with a geometry), conversion rates (to calculate the population equivalent), the municipality's operating margin for the water resource (to be compared with the water resource requests);
- **EPANET interface:** external software that the provider may use to make simulations on the distribution network to decide whether the

request may be met or not. The system generates some nodes, which are then entered into the EPANET distribution network file;

- **Shared quality:** when installed, the system has *5 inbuilt quality levels* (excellent, good, adequate, low, poor) and *2 inbuilt breakdown levels* (water leak, general breakdown). When a quality or breakdown level and a point are selected on the map, the system saves the level/geo-referenced point and forwards it to the other users;
- **Geographic information:** the geographic information contained in the GeoServer may be divided into two macro-categories: *background data* (which concern the representation of an area: OpenStreetMap, Orthophotographs 10K, CTR 10K and 2K) and *overlay data* (municipal boundaries, servicing areas, geometries, distribution network, sources, plants). Features from the OpenLayers library are used;
- **Log & Feedback:** the system emails the administrator about a failure that has occurred. However, there are instances in which the system fails to automatically notify a failure (mainly because user interfaces may be 'sensitive' to different browsers or different releases). Because of this, a user-friendly but powerful feedback-forwarding mechanism has been developed for troubleshooting purposes;
- **Internationalisation:** the system is available in Italian and English. Any language may be added on condition the translation files are supplied;
- **Configuration:** all configuration parameters of the portal reside in the config/params.php file and may be used to select the behaviour of some system components and easily change it, as needed. The specifications of all configured parameters are detailed in the deliverable.

## 2 Sommario esecutivo

Il **D11.2 – Motore WIZ (WIZ Engine)** contiene le informazioni relative all'architettura del motore alla base della piattaforma on-line sviluppata per i servizi WIZ4Planner e WIZ4All.

Il motore WIZ è ora disponibile e scaricabile da <http://wiz.acque.net/index.php?r=site/page&view=source> senza restrizioni.

Il documento si apre con una descrizione del FrameWork di base, ovvero la struttura di supporto su cui un software può essere organizzato e progettato, e del patrimonio informativo di fondo.

Seguono le specifiche sull'architettura del sistema.

La stesura del documento è frutto di una azione di confronto tra Ingegnerie Toscane Srl, beneficiario responsabile dell'Azione #11 (Implementazione del motore di proiezione WIZ), e il fornitore esterno incaricato dello sviluppo del sistema, nonché degli altri partner di progetto.

### 3 Scopo

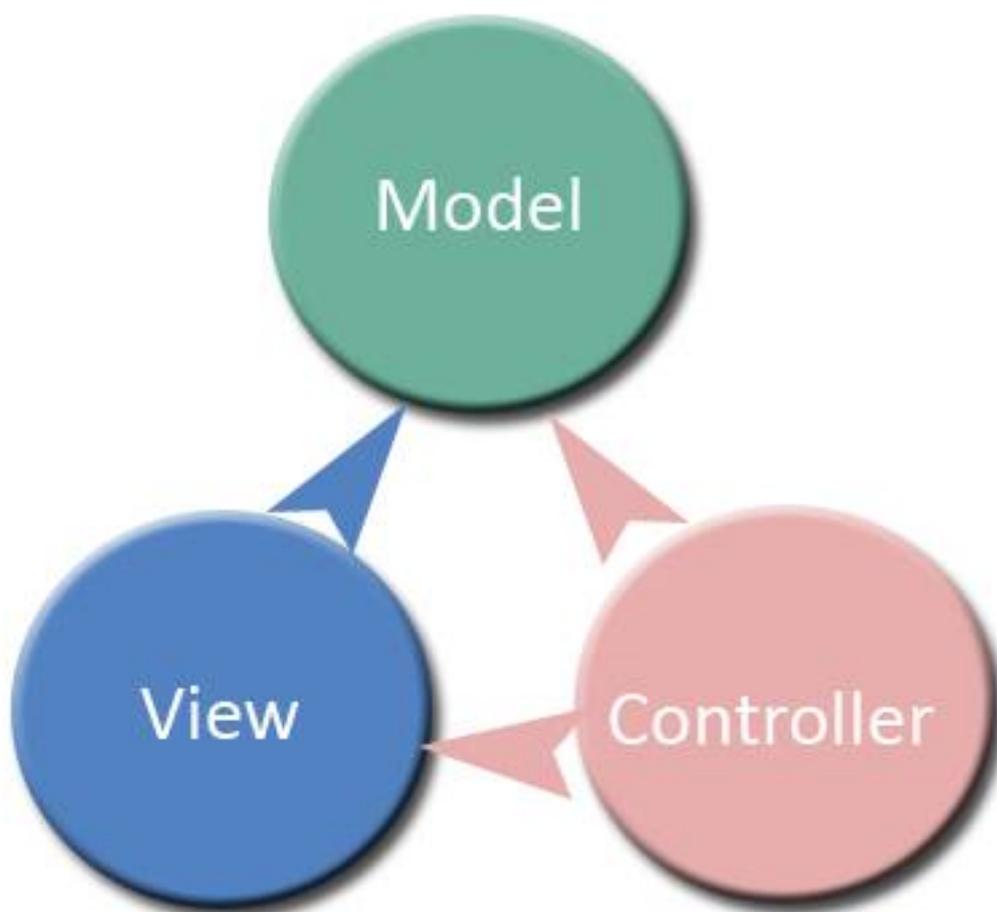
Lo scopo del deliverable **D11.2 – Motore WIZ (WIZ Engine)** è quello di definire le componenti dell'architettura del cuore della piattaforma WIZ, impalcatura necessaria per l'agganciamento delle applicazioni di livello superiore.

### 4 Framework

Il portale WIZ si basa su di un framework opensource che si chiama Yii (<http://www.yiiframework.com/>).

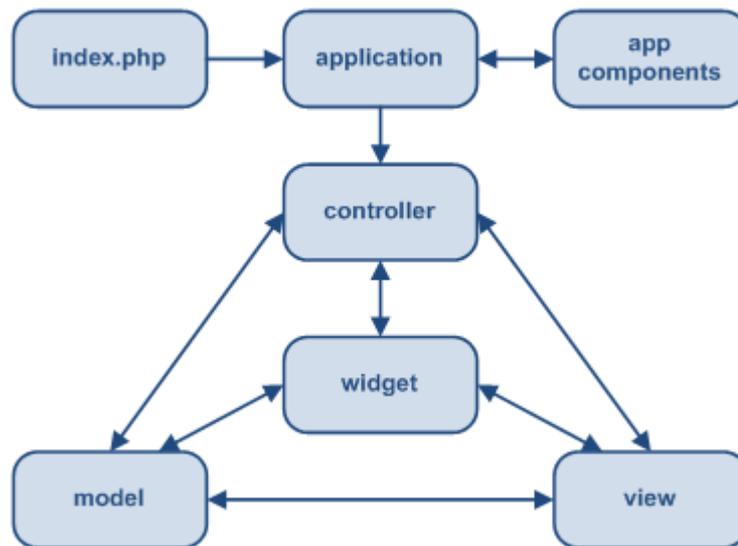
Yii è un framework per la programmazione web e può essere impiegato per lo sviluppo di qualsiasi tipo di applicazione web. È scritto in PHP e, grazie a sofisticati meccanismi di caching, risulta molto performante specialmente per le applicazioni con elevato traffico.

Yii implementa il design pattern model-view-controller (MVC), ampiamente utilizzato nella programmazione web. L'MVC consente di separare la login dall'interfaccia utente, permettendo agli sviluppatori di modificare facilmente la prima senza avere effetti sull'altra.



*Illustration 1: MVC*

Nell'MVC, il modello è rappresentato dalle informazioni (cioè dai dati), le viste contengono elementi dell'interfaccia utente come il testo o i campi di input e il controller gestisce le comunicazioni tra il modello e le viste. Il diagramma di Figura 2 illustra la struttura di un'applicazione Yii.



*Illustration 2: Applicazione Yii*

Il controller viene creato ad ogni invocazione dell'utente ed esegue l'azione richiesta, utilizzando i dati memorizzati nel modello e visualizzandoli tramite una vista appropriata.

Il modello rappresenta un singolo dato. Può essere una riga di una tabella del database o un campo di input di una form. Ogni campo del dato è rappresentato da un attributo del modello.

La vista è uno script in PHP che contiene elementi dell'interfaccia utente, come testo o campi di input. Sebbene la vista, oltre a mostrare dei dati, può anche effettuare delle operazioni su di essi, è consigliato effettuare queste operazioni sul controller, in modo da avere una netta separazione tra la logica e l'interfaccia.

## 5 Il patrimonio informativo

Per il funzionamento del sistema sono necessari una serie di dati di contorno, geografici e non; alcuni di questi popoleranno fisicamente il sistema mentre altri saranno utilizzati a richiesta accedendo a servizi W\*S secondo lo standard OGC.

L'insieme minimo dei dati necessari richiesti è composta da:

- Limiti amministrativi: riporta, in formato shape, i limiti amministrativi dei comuni gestiti da Acque SPA.
- Aree di servizio: riporta, in formato shape, i contorni delle aree di servizio nelle quali viene suddiviso ogni comune.
- Tabella della disponibilità: indica, sia per i comuni che per le aree di servizio, qual è il margine operativo attuale ed eventuale uno o più margini futuri

- Tabella dei consumi: indica, per ogni comune, qual è il consumo medio annuo e massimo giornaliero della risorsa idrica
- DEM: modello di elevazione del terreno

Questi dati risiedono fisicamente sul sistema; la loro copertura territoriale deve essere almeno pari alla porzione di territorio che Acque SPA gestisce.

Per rendere il sistema più appetibile e fornire maggiori informazioni agli utenti che lo utilizzano, sono anche richiesti:

- Ortofoto: fotografie aeree georeferenziate. Vengono caricate a richiesta sfruttando il servizio GEOscopio offerto dalla Regione Toscana
- CTR: carta tecnica regionale. Viene caricata a richiesta sfruttando il servizio GEOscopio offerto dalla Regione Toscana
- Rete di distribuzione: contiene informazioni sulla rete di distribuzione dell'acqua, georeferenzata, con le sue caratteristiche tecniche e di servizio
- Fonti: mostra l'esatta ubicazione delle fonti, suddivise in:
  - corsi d'acqua
  - laghi
  - pozzi
  - sorgenti
- Impianti: mostra l'esatta ubicazione degli impianti, suddivisi in:
  - potabilizzazione
  - adduttrici
  - accumulo
  - pompaggio
- Stradario: nomi di strade, piazze e luoghi, georeferenzati.

Le Ortofoto e la CTR sono dati esterni che vengono caricati a richiesta sfruttando il servizio GEOscopio offerto dalla Regione Toscana; tutti gli altri dati, invece, risiedono fisicamente sul sistema. Essendo dati di contorno e, quindi, non indispensabili per il corretto funzionamento del sistema, la loro copertura può anche essere inferiore alla porzione di territorio che Acque SPA gestisce; tuttavia, è auspicabile che anche per questi dati la copertura sia almeno pari a quella dei comuni gestiti da Acque SPA.

## 6 L'ARCHITETTURA

Il sistema è stato progettato per interagire con una base di dati geografica. I dati vengono memorizzati in un database Postgres con l'estensione PostGIS.

Per visualizzare le informazioni geografiche e renderle disponibili a terzi tramite standard OGC utilizzando servizi WFS/WMS si utilizza un server di mappe, GeoServer.

L'architettura viene completata da un insieme di librerie javascript opensource che permettono di interrogare facilmente il server di mappe ed aggiungere funzionalità all'interfaccia utente. In Figura 3 è rappresentato uno schema di massima dell'architettura.

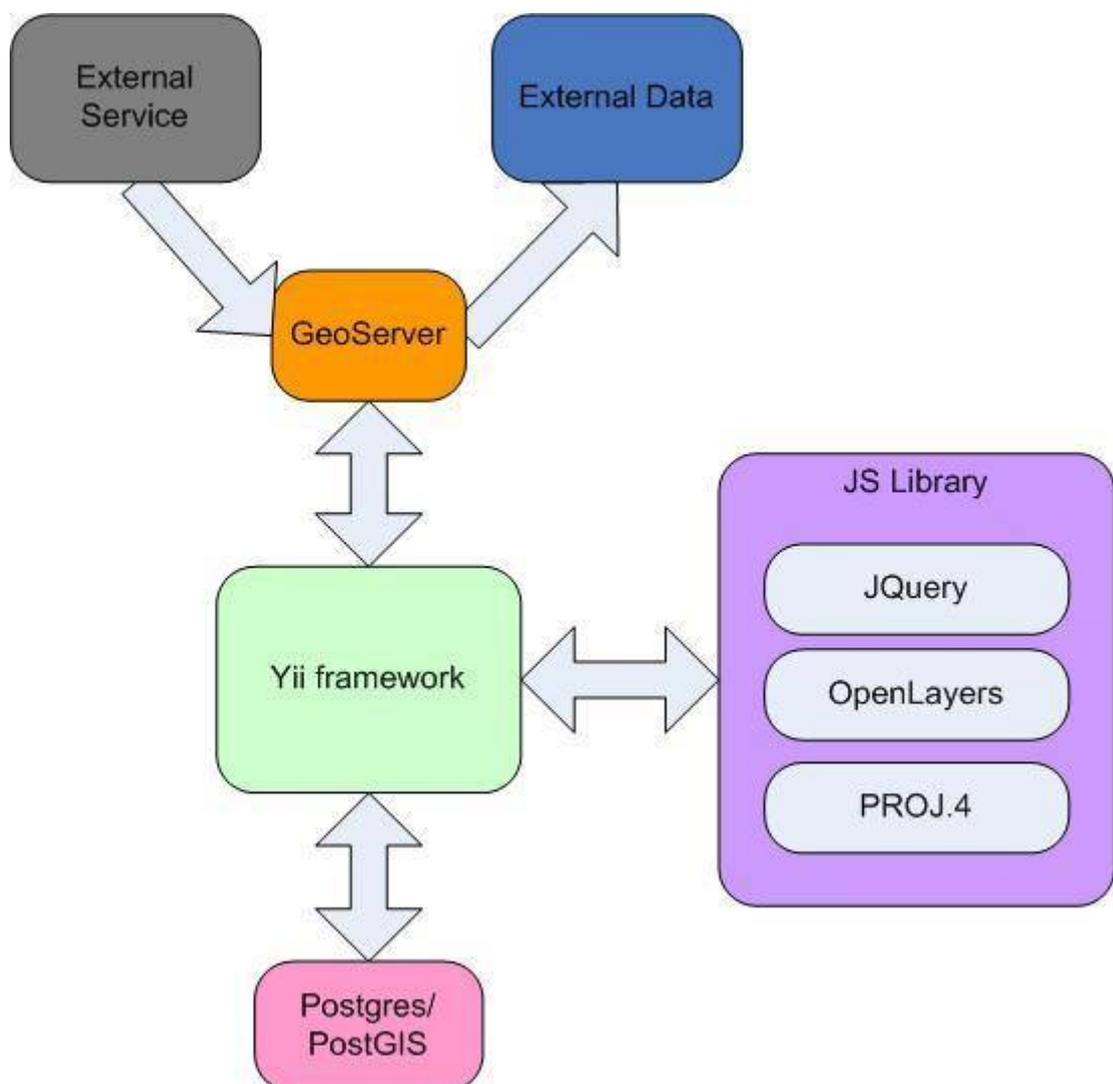


Illustration 3: Architettura

Ogni volta che si presenta una richiesta, il sistema esegue delle elaborazioni coinvolgendo tutti o parte dei componenti dell'architettura.

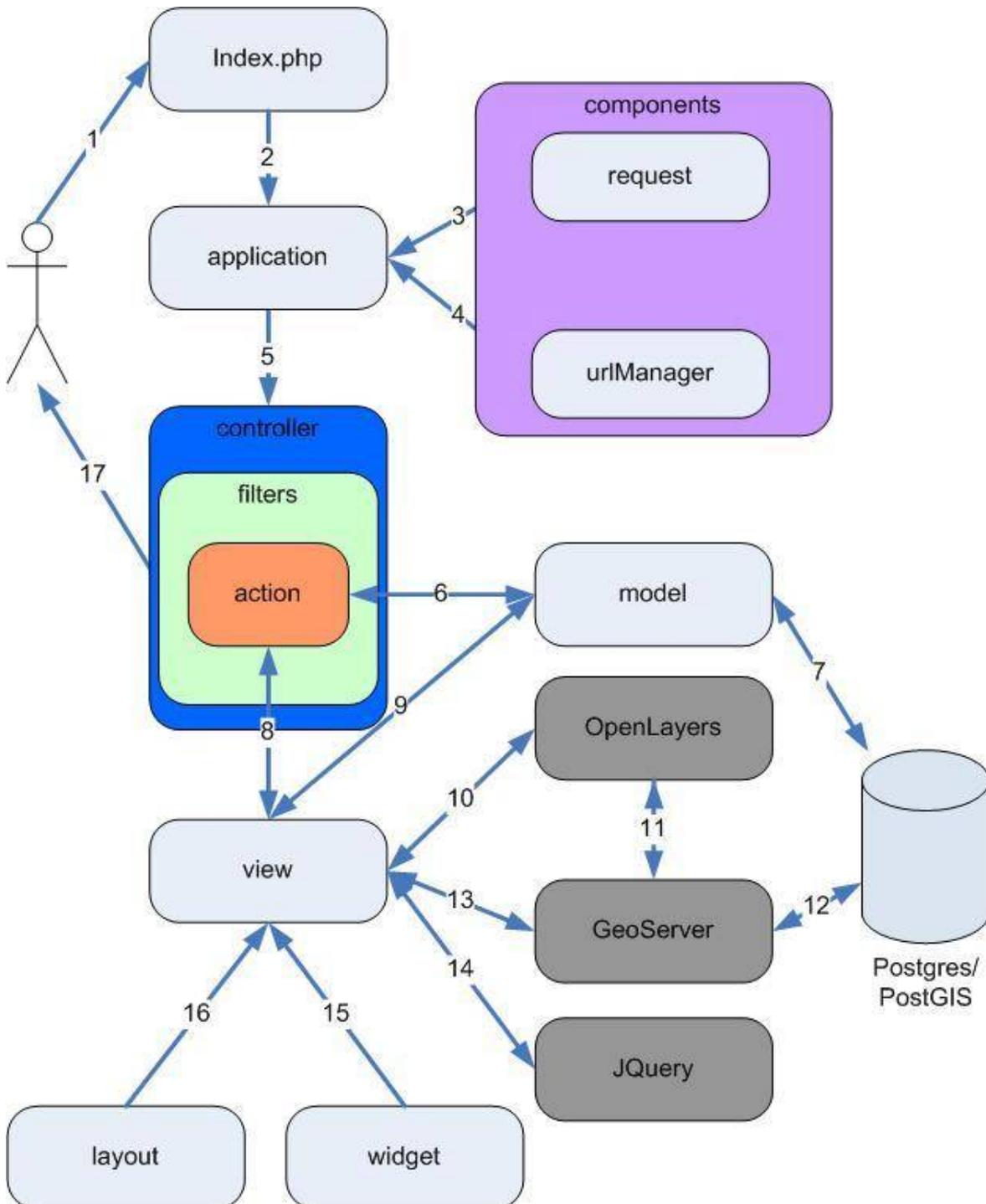


Illustration 4: Workflow dell'applicazione

Facendo riferimento allo schema di Figura 4:

1. L'utente richiede una pagina

2. Viene creata un'istanza di 'application'
3. application ottiene informazioni sulla richiesta da 'request'
4. application determina il controller da invocare grazie a 'urlmanager'
5. Il controller determina la action da utilizzare (dopo aver invocato gli opportuni filtri)
6. action legge i dati dal model
7. model recupera i dati dal DB
8. action renderizza la view appropriata
9. view legge i dati di model
10. view invoca funzioni di OpenLayers
11. si interfaccia con GeoServer
12. GeoServer recupera i dati da DB
13. view invoca direttamente GeoServer
14. view invoca funzioni JQuery
15. view esegue delle widget
16. view renderizza il risultato in un layout
17. il risultato viene mostrato all'utente

## **6.1 Il Database**

WIZ utilizza un database con estensione spaziale per memorizzare i dati; si è scelto di utilizzare, come DBMS, Postgres con l'estensione PostGIS. Il database contiene sia tabelle geografiche che non; in Figura 5 è riportato lo schema completo del database e le relazioni tra le varie tabelle.



- il tipo di operazione che genera la notifica
- il ruolo destinatario della notifica

#### **6.1.1.2 settings**

Memorizza le impostazioni sull'invio delle mail per i vari utenti. Ogni setting è caratterizzato da:

- lo username dell'utente al quale si riferisce il setting
- una categoria di notifica
- una flag che indica se, data la categoria di notifica, inviare o meno la mail

#### **6.1.1.3 users**

è la tabella degli utenti del sistema. Ogni utente è caratterizzato da:  
uno username univoco

- un nome
- un cognome
- un comune di appartenenza
- un'organizzazione
- un titolo
- un indirizzo email
- una password
- un flag che indica se l'account è attivo
- un ruolo
- un flag che indica se l'account è stato approvato
- un timestamp relativo all'ultimo login
- un link per l'attivazione veloce dell'account

#### **6.1.1.4 roles**

Memorizza i ruoli utente disponibili. Ogni ruolo è caratterizzato da:

- un nome univoco
- una descrizione
- un flag che indica se il ruolo è attivo e, quindi, può essere assegnato ad un utente
- un flag che indica se il ruolo può essere autonomamente scelto dall'utente in fase di registrazione
- un flag che indica se è possibile ricercare utenti con quel determinato ruolo

### **6.1.1.5**    *water\_qualities*

Memorizza i livelli di qualità che un utente può esprimere. Ogni livello è caratterizzato da:

- un id univoco
- una descrizione di qualità
- un colore per la visualizzazione
- una priorità per l'ordine di visualizzazione

### **6.1.1.6**    *water\_quality\_options*

Memorizza le opinioni di qualità inviate dagli utenti. Ogni opinione è caratterizzata da:

- un id univoco
- lo username dell'utente che ha espresso l'opinione
- un timestamp
- la qualità espressa
- un punto (geografico) di riferimento

### **6.1.1.7**    *water\_faults*

Memorizza i livelli di guasto che un utente può esprimere. Ogni livello è caratterizzato da:

- un id univoco
- una descrizione di guasto
- un colore per la visualizzazione
- una priorità per l'ordine di visualizzazione

### **6.1.1.8**    *water\_fault\_options*

Memorizza i guasti inviati dagli utenti. Ogni guasto è caratterizzata da:

- un id univoco
- lo username dell'utente che ha indicato il guasto
- un timestamp
- il tipo di guasto
- un punto (geografico) di riferimento

### **6.1.1.9**    *water\_requests*

Memorizza le richieste di risorsa idrica inviate dagli utenti. Ogni richiesta è caratterizzata da:

- un id univoco
- un timestamp indicante la data di creazione della richiesta

- il nome del progetto
- una descrizione
- uno stato
- una fase
- un collegamento ad una richiesta padre (se esiste)
- le note
- l'idroesigenza totale
- l'idroesigenza effettiva

#### **6.1.1.10 *water\_request\_history***

Tiene traccia delle transizioni di stato di una richiesta di risorsa idrica. Ogni transizione è caratterizzata da:

- l'id della richiesta
- un timestamp in cui avviene la transizione
- lo stato finale
- un commento

#### **6.1.1.11 *water\_request\_geometries***

Memorizza le geometrie che compongono una richiesta di risorsa idrica. Ogni geometria è caratterizzata da:

- un id univoco
- un insieme di punti (georeferenziati) che la caratterizzano
- l'id della richiesta alla quale appartiene
- un nome
- l'idroesigenza associata

#### **6.1.1.12 *water\_request\_geometry\_zones***

Memorizza le destinazioni d'uso relative ad un geometria. Ogni destinazione d'uso è caratterizzata da:

- un id univoco
- l'id della geometria alla quale fa riferimento
- la destinazione d'uso
- gli abitanti equivalenti
- l'idroesigenza associata

#### **6.1.1.13 *water\_request\_geometry\_zone\_properties***

Memorizza le coppie parametro-valore che caratterizzano una destinazione d'uso. Ogni coppia è caratterizzata da:

- un id univoco
- la destinazione d'uso di riferimento
- un parametro
- un valore associato al parametro

#### **6.1.1.14 *water\_request\_parameters***

Contiene l'elenco dei parametri disponibili associabili ad una destinazione d'uso. Ogni parametro è caratterizzato da:

- un nome
- una descrizione
- un'unità di misura

#### **6.1.1.15 *zones***

Contiene l'elenco di tutte le destinazioni d'uso disponibili. Ogni destinazione d'uso è caratterizzata da:

- un nome
- una descrizione
- un flag che indica se la destinazione d'uso è attiva, e quindi può essere selezionata
- un flag che indica se la destinazione d'uso è ricercabile
- una destinazione d'uso padre (se esiste)

#### **6.1.1.16 *zones\_water\_request\_parameters***

Indica, per ogni destinazione d'uso, quali sono i parametri che devono essere richiesti. Ogni parametro è caratterizzato da:

- un nome
- una destinazione d'uso
- un indice di conversione
- un flag che indica se il parametro è attivo o meno

#### **6.1.1.17 *water\_demand\_formulas***

Memorizza le formule necessarie per il calcolo dell'idroesigenza. Ogni formula è caratterizzata da:

- la destinazione d'uso per la quale si applica la formula
- la formula

#### **6.1.1.18 *water\_supply***

Memorizza i consumi di risorsa idrica relativi ad ogni comune. Ogni consumo è caratterizzato da:

- un id univoco

- il comune al quale si riferisce
- il consumo massimo giornaliero (in l/s)
- il consumo medio annuo (in l/s)

#### **6.1.1.19 city\_state\_boundaries**

Contiene la lista dei comuni. Ogni comune è caratterizzato da:

- un id univoco
- un nome
- in insieme di punti georeferenziati che rappresentano i confini

#### **6.1.1.20 operative\_margin**

Indica, per ogni comune, qual è il margine operativo della risorsa ad oggi e quale sarà, invece, in futuro. Ogni margine è caratterizzato da:

- un id univoco
- il comune al quale si riferisce
- il margine (in l/s)
- una descrizione di scenario per il margine (vuoto se il margine di riferisce alla situazione attuale)

#### **6.1.1.21 service\_area**

Contiene la lista delle aree di servizio. Ogni area di servizio è caratterizzata da:

- un id univoco
- un nome
- una descrizione
- un flag indicante se l'area di servizio è reale o virtuale
- l'area in metri quadrati
- in insieme di punti georeferenziati che rappresentano i confini

#### **6.1.1.22 service\_area\_operative\_margin**

Indica, per ogni area di servizio, qual è il margine operativo della risorsa ad oggi e quale sarà, invece, in futuro. Ogni margine è caratterizzato da:

- un id univoco
- l'area di servizio alla quale si riferisce
- il margine (in l/s)
- una descrizione di scenario per il margine (vuoto se il margine di riferisce alla situazione attuale)

### **6.1.1.23 plugins**

Memorizza i plugins installati sul sistema. Ogni plugin è caratterizzato da:

- un nome univoco
- il nome dell'utente che l'ha installato
- il timestamp dell'installazione
- la versione
- una priorità in base alla quale verrà visualizzato nel menù
- gli utenti che lo possono utilizzare
- lo stato dell'installazione
- un flag che indica se il plugin è attivo
- una descrizione
- un changelog

### **6.1.1.24 sendmail\_logs**

Memorizza il log relativo all'invio delle e-mail. Il log è caratterizzato da:

- un id univoco
- un timestamp dell'invio
- la categoria della mail inviata (notifica, segnalazione, ecc)
- un riferimento all'entità che ha generato l'invio della mail
- il risultato dell'operazione di invio (successo/errore)
- l'eventuale codice di errore
- l'eventuale messaggio associato all'errore
- il destinatario della mail (un utente oppure tutti gli utenti appartenenti ad un determinato ruolo)

## **6.2 Controllo degli accessi**

Il controllo degli accessi è piuttosto sofisticato e si basa sul concetto di autorizzazione. L'autorizzazione, cioè il permesso di un utente per compiere una determinata operazione, può essere:

- un ruolo
- un task
- un'operazione

Un ruolo è composto da diversi task, ogni task è composto da diverse operazioni e un'operazione è un permesso che è atomico.

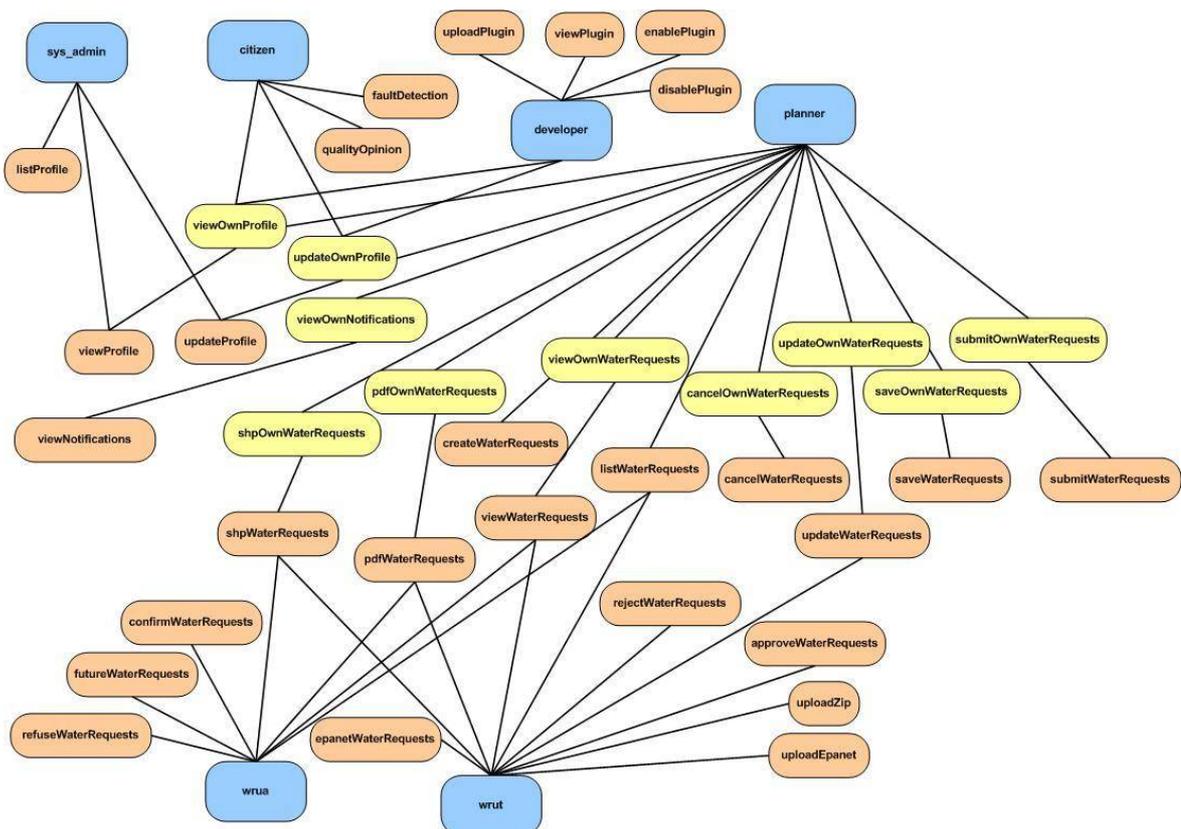
Queste entità solo, quindi, collegate tra loro tramite relazione padre-figlio; inoltre, per una maggiore flessibilità, è possibile legare tra loro,

sempre tramite relazioni padre-figlio, i ruoli, i task e le operazioni. Se una entità A è padre di un'altra entità B, allora A eredita tutti i permessi rappresentati da B.

Quello che si ottiene alla fine, piuttosto che un albero, è un grafo gerarchico parzialmente ordinato in cui nel livello superiore si trovano i ruoli e in quello inferiore le operazioni; nel mezzo i task.

Infine, sempre per avere maggiore flessibilità, è possibile associare, ad ogni entità (ruolo, task o operazione), una regola. La regola è un pezzo di codice che viene valutato ogni volta che il sistema deve stabilire se un utente ha un determinato permesso; solo l'esecuzione restituisce esito positivo l'utente avrà il permesso rappresentato dall'entità.

In Figura 6 viene mostrata il grafo degli accessi del sistema. Per semplicità sono state riportate solo le relazioni principali. I riquadri in blu rappresentano i ruoli; quelli in giallo i task; quelli in arancione le operazioni.



*Illustration 6: Grafo controllo accessi*

### 6.3 Le notifiche

Il sistema genera in automatico delle notifiche al verificarsi di certi eventi. Ogni notifica è caratterizzata da una categoria che indica:

- l'entità coinvolta nella notifica (utenti, richieste di risorsa idrica, ...)
- l'evento che genera la notifica (creazione, aggiornamento, ...)

Le notifiche possono essere indirizzate ad un solo utente oppure a tutti gli utenti appartenenti ad uno stesso ruolo.

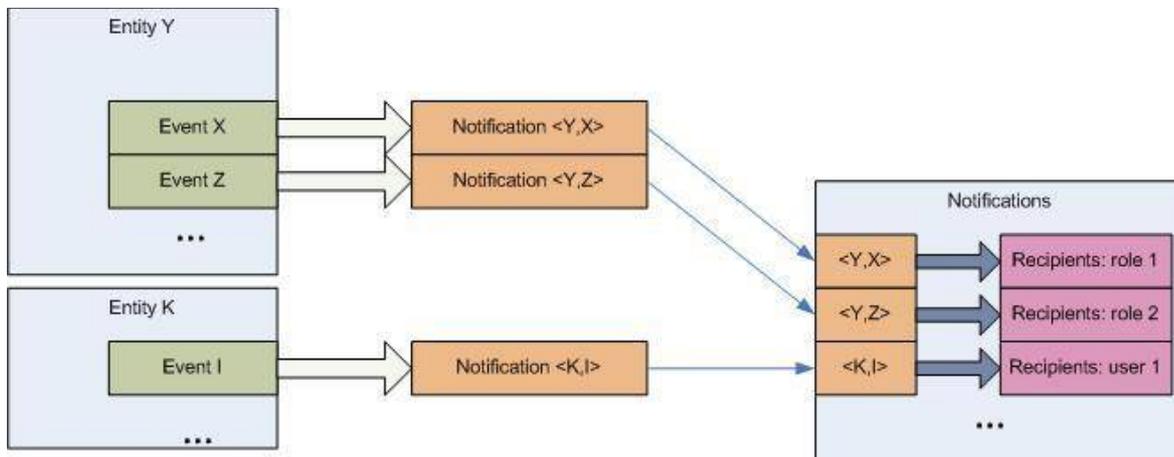


Illustration 7: Struttura delle notifiche

In questo secondo caso, la notifica è comunque unica, anche se visibile da più utenti (cioè tutti quelli appartenenti ad uno stesso ruolo). Se un utente, quindi, marca la notifica come 'letta' anche gli altri la vedranno come 'letta'. In questo caso il flag, più che come 'letta' deve intendersi come 'presa in carico'.

Ogni utente visualizza le notifiche che riceve accedendo all'omonima sezione nel portale. Il sistema inoltra il contenuto della notifica anche per e-mail all'utente o agli utenti destinatari della notifica stessa. Tuttavia, accedendo alla sezione impostazioni, ciascun utente può disabilitare l'invio della e-mail per una particolare entità o per una determinata operazione.

## 6.4 I plugins

I plugins permettono di arricchire le funzionalità del sistema. Vengono caricati dall'utente sviluppatore sotto forma di archivio; sempre l'utente sviluppatore può gestire i plugin tramite un'opportuna interfaccia utente.

Un plugin non è altro che un'entità con le sue viste, il suo modello e il suo controller; è, a tutti gli effetti un componente MVC che, quindi, si integra facilmente con il sistema. Questa struttura permette di semplificare notevolmente la procedura di installazione di nuovi plugin; è sufficiente, infatti, aggiungere ai controller, viste e modelli già esistenti quelli relativi al plugin ed, eventualmente, eseguire uno script sql (se fornito insieme al plugin). Per una migliore gestione di tutti i plugin,

viene anche richiesto di caricare informazioni sulla versione del plugin e una descrizione sulle sue funzionalità.

Chiaramente il sistema si aspetta di ricevere queste informazioni in maniera struttura; inoltre i controller, le viste e i modelli forniti devono necessariamente rispettare delle semplici regole; per maggiori informazioni sui plugins e sulla loro creazione si rimanda al documento 'Plugins ed estensioni'.

## 6.5 Gli utenti

Il sistema offre la maggior parte delle funzionalità solo dopo aver effettuato il login. Per questo motivo, un utente che intende utilizzare il sistema deve preventivamente registrarsi inserendo:

- Nome (obbligatorio)
- Cognome (obbligatorio)
- Comune di appartenenza (obbligatorio)
- Organizzazione
- Titolo
- Indirizzo email (obbligatorio)
- Username (obbligatorio)
- Password (obbligatorio)
- Ruolo (obbligatorio)

La password viene memorizzata facendo un hash md5.

Il campo ruolo permetterà all'utente di scegliersi un ruolo all'interno del sistema; la lista dei ruoli disponibili in fase di registrazione viene recuperata dalla tabella role grazie al flag on\_registration.

Tutti le registrazioni avranno il flag approved settato a falso, ad indicare che l'account deve essere preventivamente attivato dal gestore del sistema. Quindi, il ruolo che l'utente sceglie autonomamente in fase di registrare deve essere successivamente convalidato dal gestore del sistema.

Il gestore del sistema può approvare gli account utente tramite un'opportuna interfaccia oppure cliccando su di un link che riceve automaticamente per email dopo ogni registrazione. Il sistema, infatti, a seguito di una registrazione utente, genera un link univoco contenente una stringa del tipo `username_<hash_di_caratteri_casuali>` e lo invia per email al gestore del sistema. Solo il gestore del sistema può conoscere questo link e quindi, cliccando semplicemente sul link stesso attiverà l'account dell'utente in questione.

Completata la fase di registrazione, per potersi autenticare l'utente deve effettuare il login.

Le informazioni richieste in fase di login sono:

- Username (obbligatorio)
- Password (obbligatorio)

La procedura di login darà esito positivo, permettendo all'utente di accedere, solo se sono vere le seguenti condizioni:

- Esiste un utente con quello username
- La password coincide con quella memorizzata
- L'account utente non è stato bloccato (flag active settato a vero)

L'accesso avverrà con il ruolo dell'utente solo se l'account è stato approvato dal gestore del sistema; con il ruolo di cittadino in caso contrario.

La Figura 8: Procedura di login illustra la procedura di login.

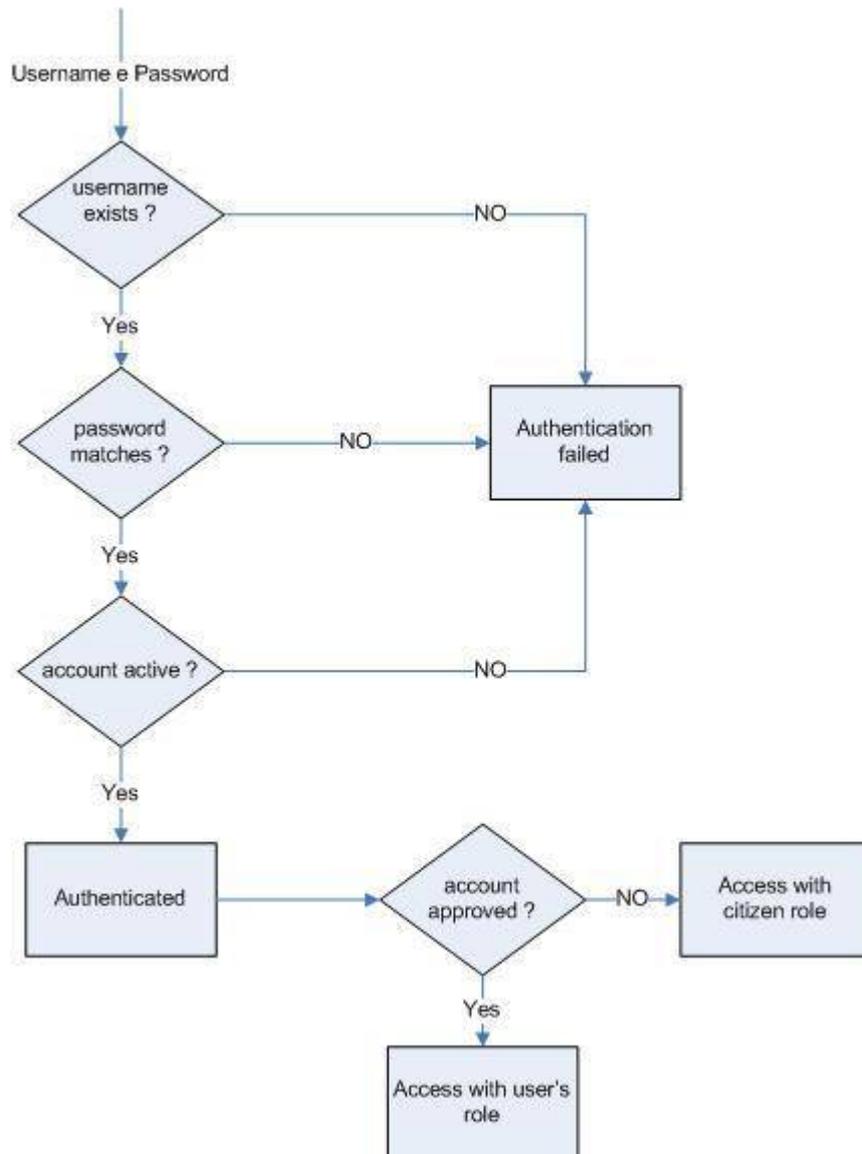


Illustration 8: Procedura di login

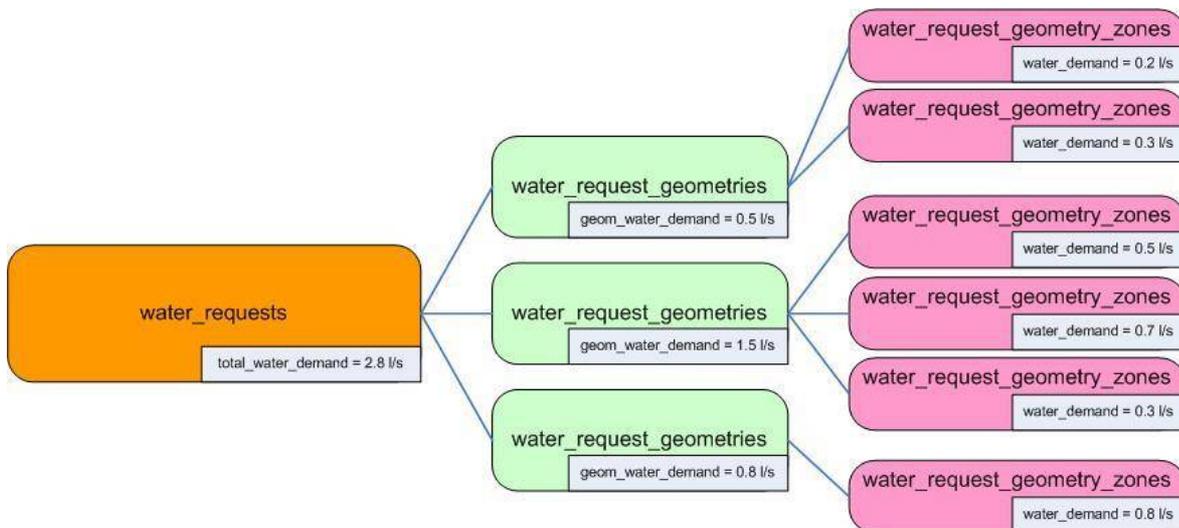
## 6.6 Le richieste di risorsa idrica

Le richieste di risorsa idrica rappresentano l'entità più complessa del sistema. È grazie a questa entità che un utente pianificatore può creare e sottomettere al gestore le proprie richieste di risorsa idrica. Esistono due tipi di richiesta di risorsa idrica e ogni richiesta evolve tra diversi stati; per maggiori informazioni sulle fasi di una richiesta e sugli stati si rimanda al documento 'Specifiche'.

Una richiesta di risorsa idrica viene memorizzata nella tabella `water_requests` e, come si vede dallo schema del database di Figura 5 coinvolge diverse entità. `water_request_geometries` contiene tutte le geometrie che compongono la richiesta. Tra le caratteristiche più importanti della geometria c'è `geom_water_demand`, che rappresenta

l'idroesigenza associata alla geometria. La somma dei vari geom\_water\_demand delle geometrie che compongono una richiesta fornisce il campo total\_water\_demand.

Ogni geometria è, a sua volta, composta da destinazioni d'uso. Le destinazioni d'uso sono memorizzate nella tabella water\_request\_geometry\_zones che riporta sia gli abitanti equivalenti che l'idroesigenza calcolata per una determinata destinazione d'uso. La somma dell'idroesigenza delle varie destinazioni d'uso all'interno di una stessa geometria fornisce il campo geom\_water\_demand.



*Illustration 9: Relazione tra water\_requests, water\_request\_geometries e water\_request\_geometry\_zones*

Il calcolo degli abitanti equivalenti e, successivamente, dell'idroesigenza, viene effettuato in automatico dal sistema e grazie all'aiuto dell'utente che dovrà rispondere ad alcune semplici domande. In Figura 10 viene illustrato il meccanismo:

1. L'utente seleziona una destinazione d'uso tra tutte quelle ammissibili, contenute nella tabella zones
2. In base alla destinazione d'uso, il sistema recupera uno o più coppie nome parametro-indice di conversione dalla tabella zones\_water\_request\_parameters
3. Per almeno un parametro proveniente dalla fase 2, l'utente dovrà inserire un valore, che verrà successivamente moltiplicato con il corrispondente indice di conversione per ottenere gli abitanti equivalenti
4. In base alla destinazione d'uso, il sistema recupera una formula dalla tabella water\_demand\_formulas che utilizzerà per il calcolo dell'idroesigenza

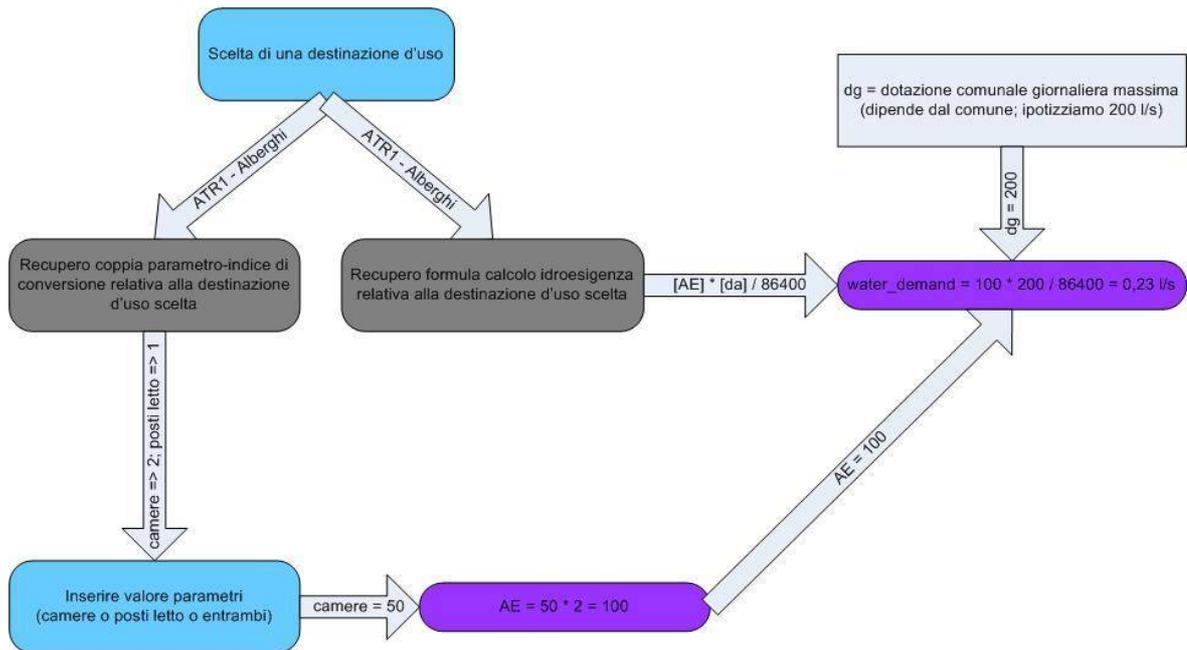


Illustration 10: Procedura per il calcolo degli abitanti equivalenti e dell'idroesigenza

Le formule, quindi, permettono di calcolare l'idroesigenza associata ad una geometria mentre gli indici di conversione servono per determinare gli abitanti equivalenti; entrambi dipendono dalla destinazione d'uso. In Figura 11 vengono mostrate le formule utilizzate dal sistema.

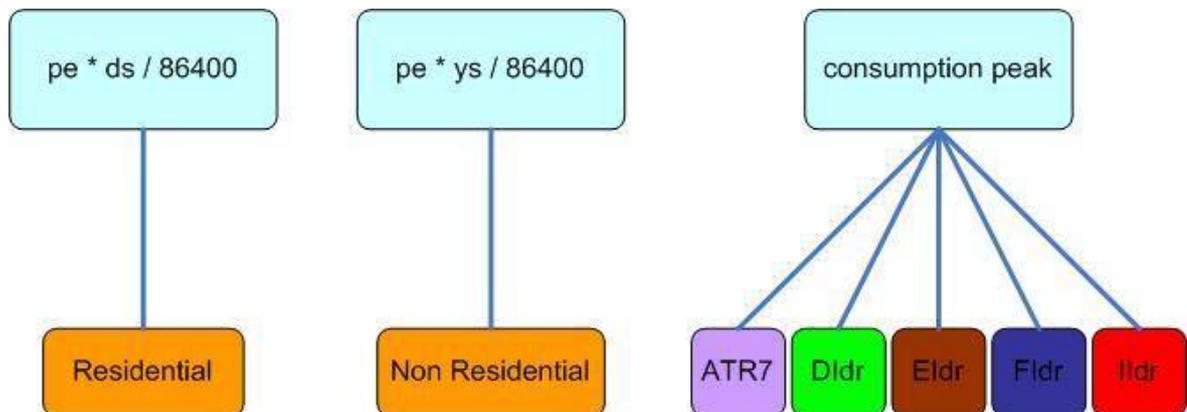
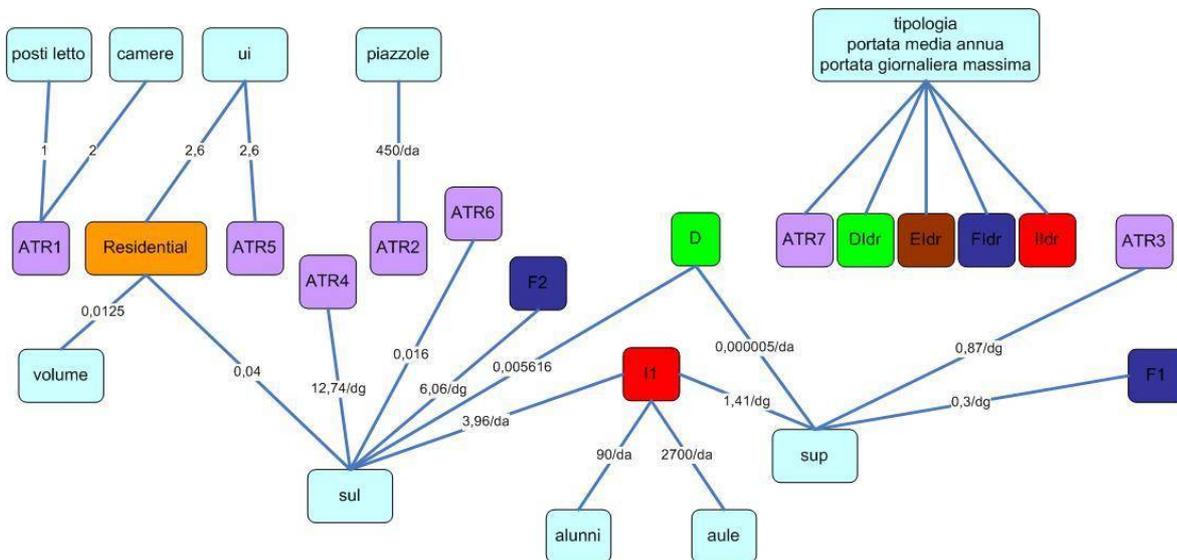


Illustration 11: Formule per il calcolo dell'idroesigenza

Gli indici di conversione vengono anche utilizzati dal sistema per porre domande mirate all'utente: data una determinata destinazione d'uso, il sistema conosce quali sono gli indici di conversione associati e, quindi, chiederà all'utente di inserire almeno uno dei parametri per i quali l'indice di conversione esiste (tralasciando i rimanenti). In Figura 12 sono riportati i parametri riconosciuti dal sistema, la loro associazione con le destinazioni d'uso ed il relativo indice di conversione.

Chiaramente uno stesso parametro può essere associato a più destinazioni d'uso (e ad ogni associazione il valore può variare) e una destinazione d'uso deve avere almeno un indice associato.



*Illustration 12: Parametri, indici di conversione e loro relazione con le destinazioni d'uso*

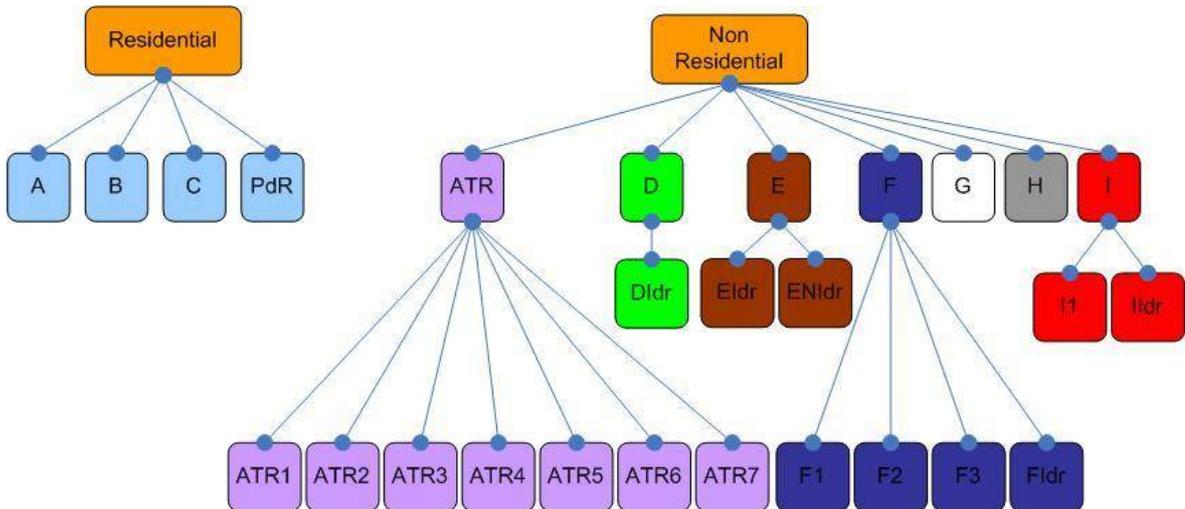
Facendo riferimento alla Figura 12, e ipotizzando come destinazione d'uso una zona 'Residenziale':

- il sistema chiederà all'utente di specificare almeno un parametro tra:
  - ui
  - sul
  - volume

perché solo per questi parametri è associato un indice di conversione relativo alla zona 'Residenziale'

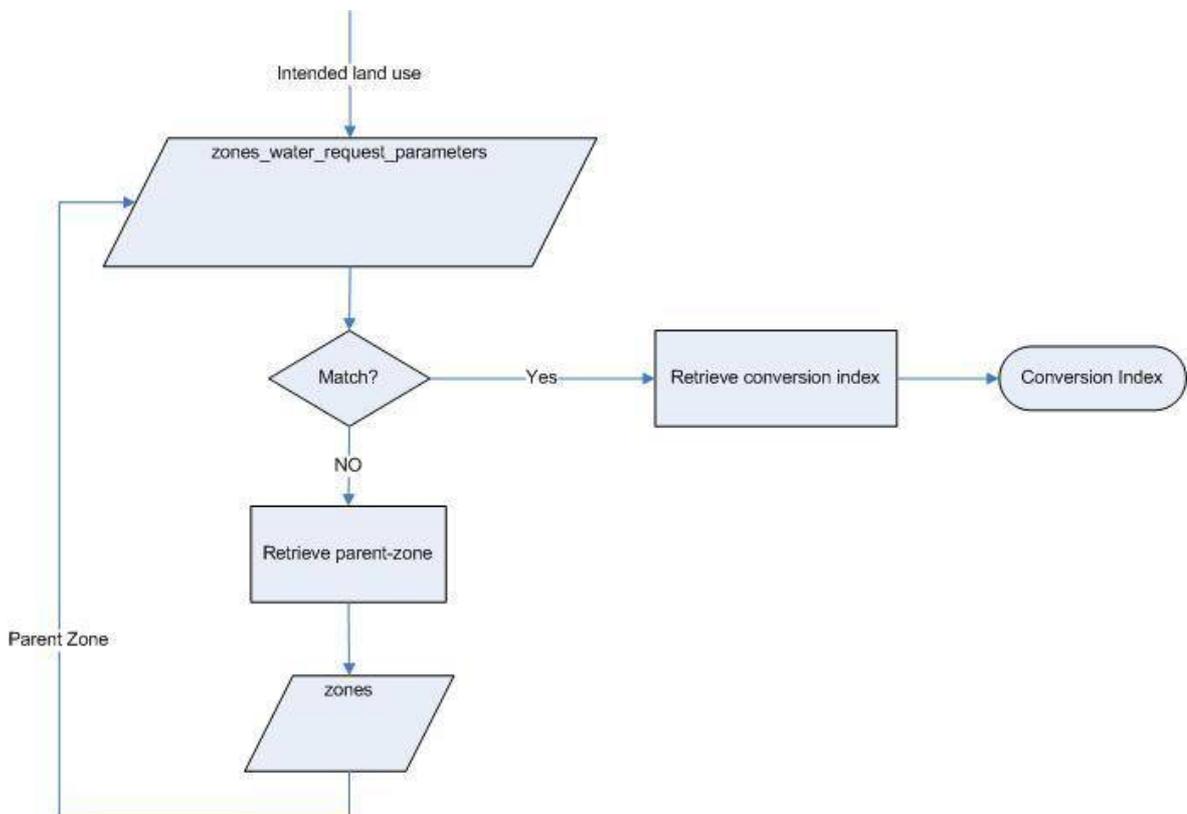
- ipotizzando che l'utente inserisca una sul di 1000m2 il sistema calcolerà 40 abitanti equivalenti ( $1000 * 0.04$ )

Per ottimizzare l'intero processo, semplificare la gestione del sistema e minimizzare i dati richiesti le destinazioni d'uso, memorizzate nella tabella zones, sono state organizzate in modo gerarchico; la Figura 13 ne mostra lo schema1.



*Illustration 13: Destinazioni d'uso organizzate gerarchicamente*

Questa struttura gerarchica viene ampiamente sfruttata nei passi 2 e 4. Infatti, quando il sistema cerca la coppia parametro-indice di conversione (passo 2 nell'elenco di pagina 29), nel caso in cui non viene trovata una corrispondenza con la destinazione d'uso scelta, il sistema tenta con la destinazione d'uso padre e così via ripercorrendo l'albero finché non si arriva in cima. In Figura 14 viene illustrata la procedura.

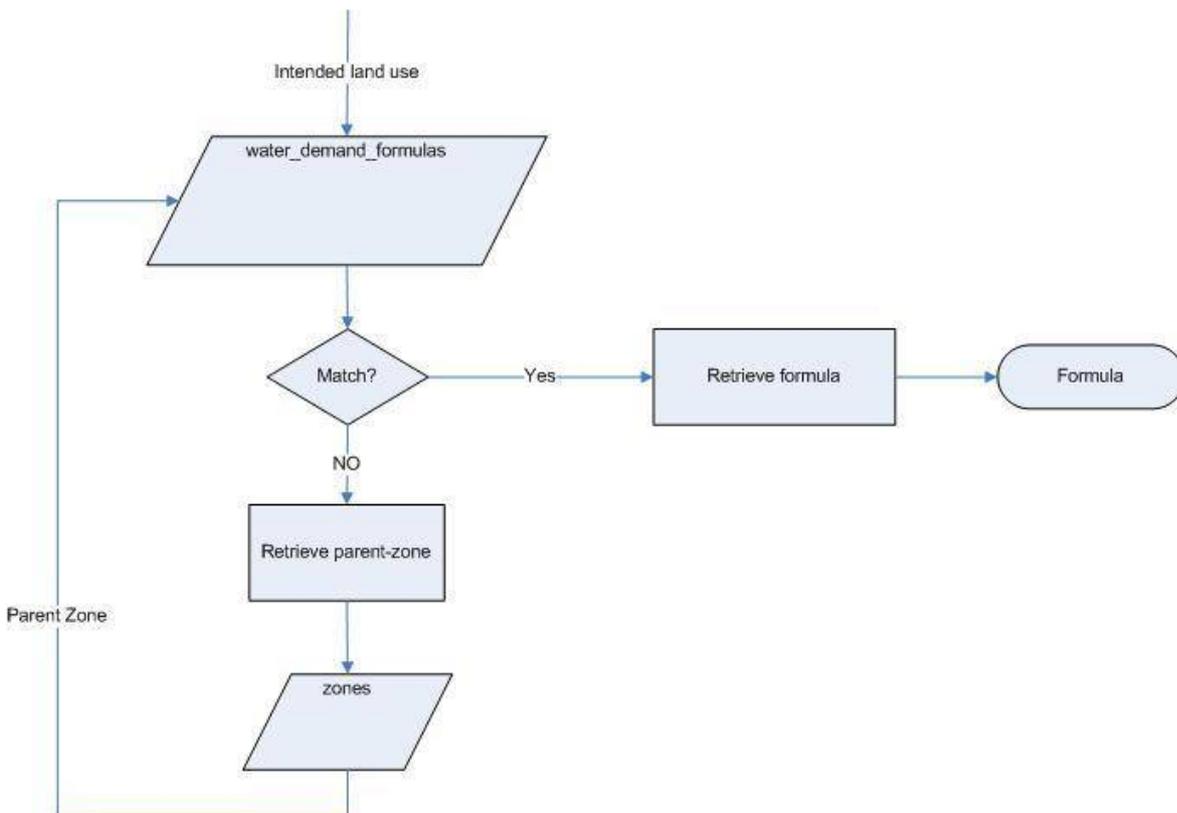


*Illustration 14: Procedura per il recupero dell'indice di conversione per il calcolo degli abitanti equivalenti*

Supponendo di aver scelto che destinazione d'uso la zona 'A – Aree Storizzate' il sistema:

1. Cerca nella tabella `zones_water_request_parameters` se esiste almeno un parametro associato alla zona 'A'
2. Se non lo trova, recupera la destinazione d'uso padre di 'A', cioè la zona 'Residenziale'
3. Ritorna nella tabella `zones_water_request_parameters` e prova a vedere se esiste almeno un parametro associato alla zona 'Residenziale'
4. Se non lo trova, torna al passo 2; altrimenti la procedura termina restituendo una o più coppie parametro-indice di conversione

Ragionamenti analoghi valgono per le formule: viene prima cercata una formula relativa alla destinazione d'uso scelta e, se non si trova, viene cercata una formula per il padre e così via. La Figura 15 mostra quest'altra procedura.



*Illustration 15: Procedura per il recupero della formula per il calcolo dell'idroresigenza*

Supponendo di aver scelto che destinazione d'uso la zona 'ATR1 – Alberghi' il sistema:

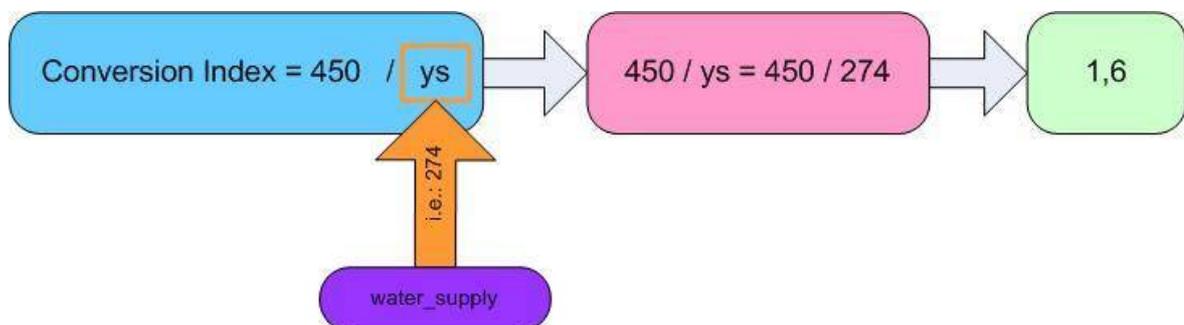
1. Cerca nella tabella `zones_water_request_parameters` se esiste almeno una formula associata alla zona 'ATR1 – Alberghi'

2. Se non lo trova, recupera la destinazione d'uso padre di 'ATR1 – Alberghi', cioè la zona 'ATR – Aree Turistico Ricettive'
3. Ritorna nella tabella `zones_water_request_parameters` e prova a vedere se esiste almeno una formula associata alla zona 'ATR – Aree Turistico Ricettive'
4. Se non lo trova, torna al passo 2; altrimenti la procedura termina restituendo la formula richiesta

Gli indici di conversione e le formule non sono dei semplici numeri ma, in generale, sono delle stringhe che contengono numeri ed operatori matematici ma anche delle variabili. Il sistema effettua quindi un parsing intelligente valutando queste stringhe e restituendo un numero. Per quanto riguarda gli indici di conversione, la loro struttura è piuttosto semplice in quanto si tratta prevalentemente di numeri; in alcuni casi sono previsti degli operatori matematici e/o solo due variabili:

- da: dotazione comunale media annua
- dg: dotazione comunale massima giornaliera

Queste due variabili dipendono chiaramente dal comune interessato alla richiesta e vengono recuperate dalla tabella `water_supply`. In Figura 16 viene illustrata la procedura.



*Illustration 16: Valutazione degli indici di conversione*

Per le formule, invece, la struttura si complica in quanto, oltre per quello che è già stato detto per gli indici di conversione, è possibile utilizzare uno o più parametri che, al momento della valutazione della formula, verranno sostituiti con il valore che ha specificato l'utente in fase di inserimento. Per chiarire questo concetto, in Figura 17 è riportato il funzionamento.

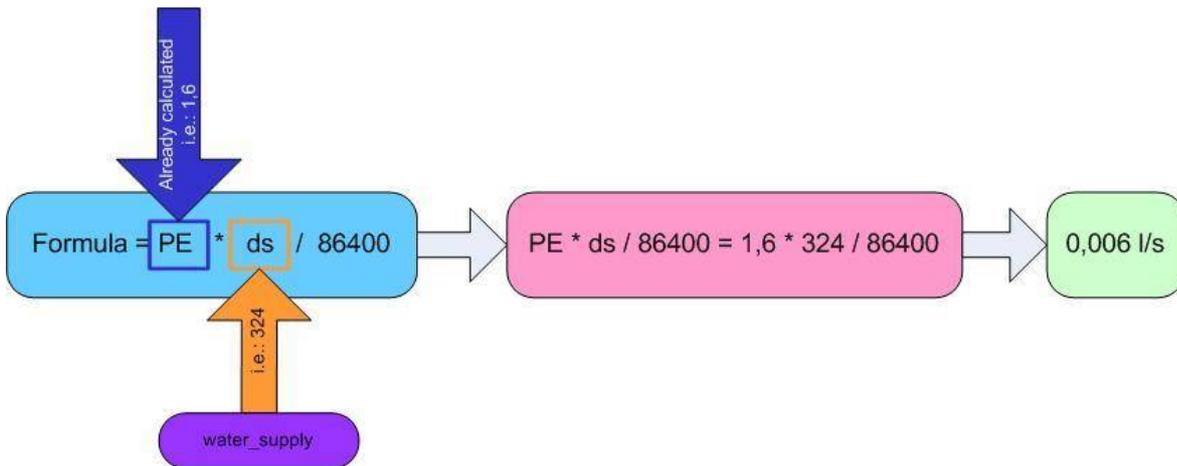


Illustration 17: Valutazione delle formule per il calcolo dell'idroesigenza

Dettagliando lo schema di Figura 10, in Errore: sorgente del riferimento non trovata viene illustrato e riassunto l'intero processo e le tabelle coinvolte per il calcolo dell'idroesigenza.

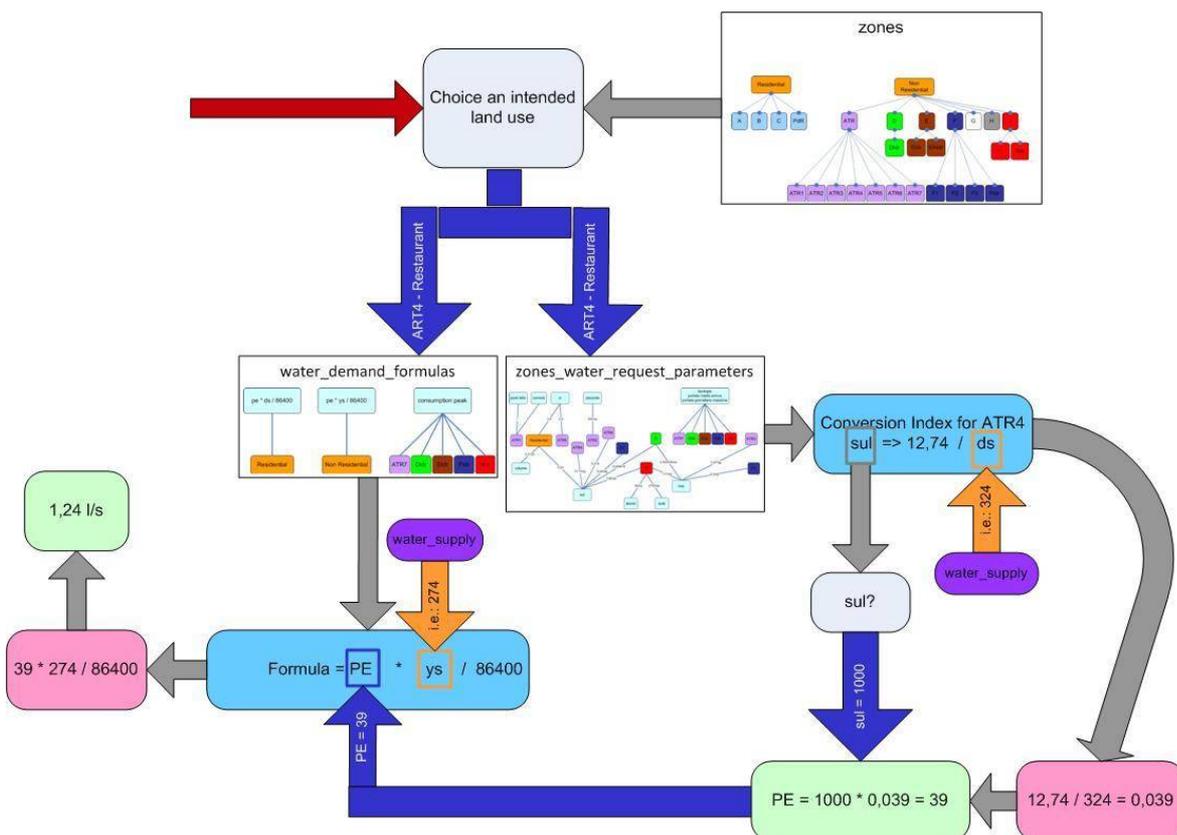


Illustration 18: Dettaglio calcolo idroesigenza

Ricavata l'idroesigenza, il sistema riesce anche a fare un controllo in tempo reale confrontando la richiesta di risorsa idrica con il margine operativo del comune, memorizzato nella tabella operative\_margin. Inoltre, sempre nella stessa tabella, sono memorizzati (se esistono) scenari di previsione futura sull'andamento della risorsa.

Per dettagliare meno la risposta, ogni comune è stato diviso in sottoaree, che prendono il nome di aree di servizio. Ogni area di servizio, oltre ad un nome ed ai confini geografici, è caratterizzata anch'essa da un margine operativo, memorizzato nella tabella `service_area_operative_margin` che contiene anche gli scenari di previsione futura (se esistono).

Ogni volta che l'utente disegna una geometria all'interno di una richiesta, il sistema calcola qual è il comune coinvolto ed in quale area di servizio cade; queste due informazioni vengono utilizzate, oltre che per i calcoli come abbiamo appena visto, anche per mostrare all'utente il margine attuale e futuro della risorsa secondo i seguenti passaggi:

- il sistema cerca se esiste un margine associato all'area di servizio interessata
- se esiste lo mostra; altrimenti utilizza il margine del comune coinvolto

Come si evince dalle procedure finora descritte, il processo decisionale che porta in automatico il sistema a determinare l'idroesigenza associata ad una richiesta è piuttosto complicato e coinvolge diverse tabelle del database, grazie alle quali il sistema incrocia dati geografici e non. Le principali tabelle coinvolte in questo processo sono:

- `confini_comunali`
- `service_area`
- `operative_margin`
- `service_area_operative_margin`
- `water_supply`
- `zones`
- `zones_water_request_parameters`

Il diagramma di flusso di Figura 19 descrive la procedura descritta.

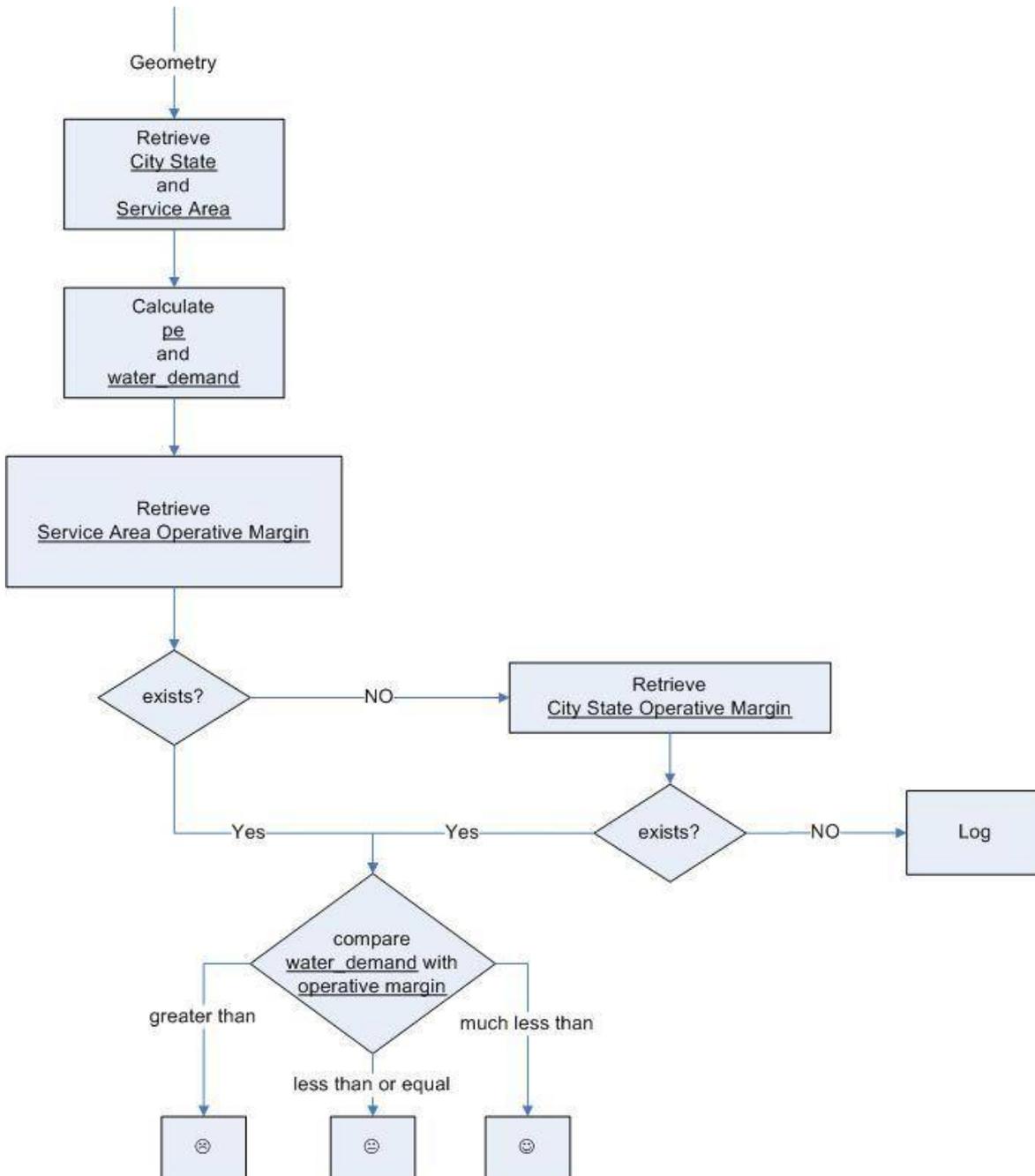


Illustration 19: Confronto tra idroesigenza e margine operativo

Tutti i valori presenti nelle tabelle coinvolte per il calcolo degli abitanti equivalenti e dell'idroesigenza sono già preventivamente impostati ma possono chiaramente essere modificati dal gestore della risorsa idrica. Oltre che modificare i dati singolarmente il gestore della risorsa idrica può caricare queste informazioni direttamente da un file .csv

## 6.7 Interfaccia EPANET

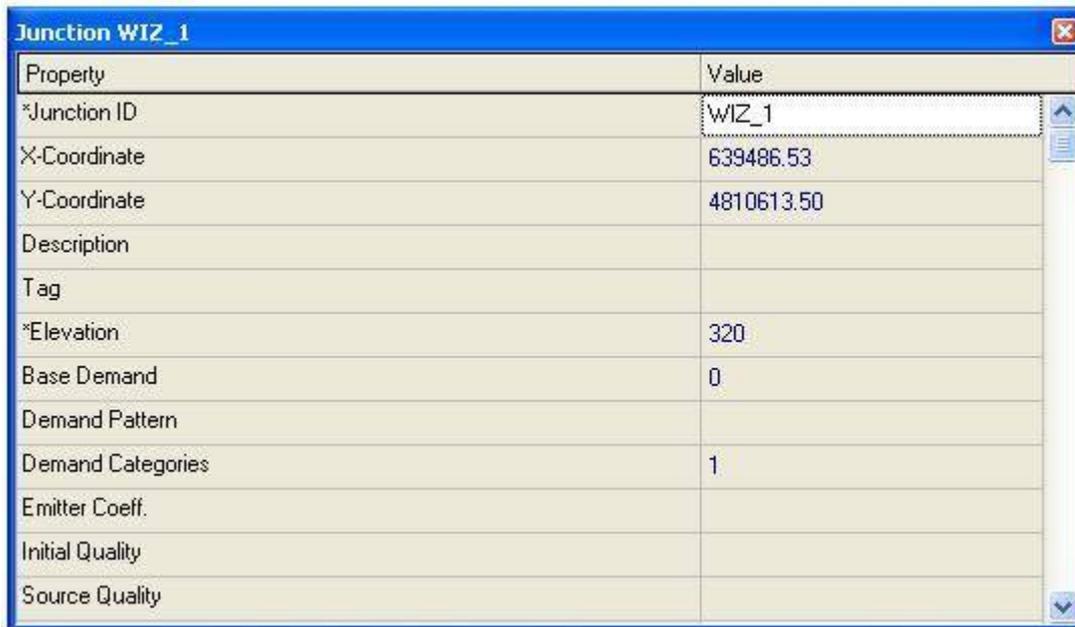
Data una richiesta di risorsa idrica, il gestore deve poter effettuare delle simulazioni sulla rete di distribuzione per decidere se la richiesta può essere soddisfatta o meno. Queste simulazioni vengono effettuate da un

software di simulazione esterno, EPANET. Il sistema deve permettere, quindi, l'interfacciamento con EPANET.

Dal punto di vista del gestore, il dato necessario per poter eseguire le simulazioni è il nuovo file della rete di distribuzione, aggiornato con i nodi (in gergo EPANET, le junctions) relativi alla richiesta di risorsa idrica in esame. Un nodo non è altro che un punto georeferenziato che corrisponde al centroide della geometria che compone la richiesta di risorsa idrica. Essendo che una richiesta può contenere più geometrie, in generale verranno generati uno o più nodi relativi alla stessa richiesta.

All'interno di EPANET, un nodo è caratterizzato da una serie di informazioni, quali:

- junction id
- x-coordinate
- y-coordinate
- description
- tag
- elevation
- base demand
- demand pattern
- demand categories
- emitter coeff.
- initial quality
- source quality



Property	Value
*Junction ID	WIZ_1
X-Coordinate	639486.53
Y-Coordinate	4810613.50
Description	
Tag	
*Elevation	320
Base Demand	0
Demand Pattern	
Demand Categories	1
Emitter Coeff.	
Initial Quality	
Source Quality	

*Illustration 20: Campi di una Junction, secondo la specifica EPANET*

Alcuni di questi parametri vengono generati automaticamente dal sistema mentre altri devono essere inserite dall'utente (e sono opzionali). In particolare, per ogni nodo, il sistema genera:

- identificatore univoco, nella forma 'WIZ\_<id della richiesta>\_<id della geometria>' (campo junction\_id)
- destinazione d'uso della geometria(campo description)
- coordinate x,y dei centroidi della geometria(campo x-coordinate e y-coordinate)
- altitudine dei centroidi della geometria(campo elevation)
- idroesigenza associata alla geometria (campo base demand)

I nodi generati dal sistema devono essere successivamente inseriti all'interno del file EPANET della rete di distribuzione. Per effettuare questa operazione è necessario che l'utente carichi il file EPANET della rete di distribuzione in modo tale che il sistema lo possa aggiornare.

Secondo le specifiche EPANET, il file di modello della rete è un file di testo, suddiviso in sezioni. Il sistema legge il file, individua le sezioni e, solo per le sezioni che interessano le junctions inserisce le informazioni necessarie. In Figura 21 è mostrato un esempio di file EPANET, nel quale, per semplicità, è riportata solo una junction con id 'WIZ\_13\_10'.

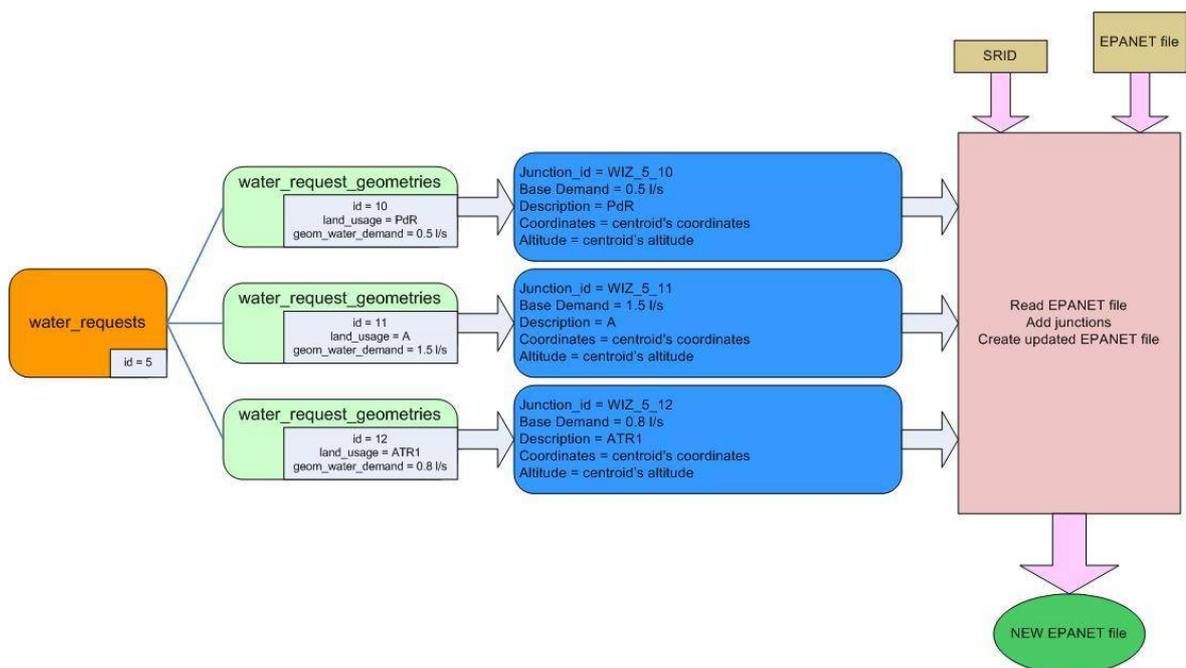
```

1 [TITLE]
2 Something
3
4 [JUNCTIONS]
5 ;ID Elev Demand Pattern
6 WIZ_13_10 94.0 2.72638888888889 ;PdR - Piano di Recupero Residenziale
7
8 [TAGS]
9 NODE WIZ_13_10 A tag
10
11 [DEMANDS]
12 ;Junction Demand Pattern Category
13 WIZ_13_10 0.04597000000 A pattern ;
14
15 [EMITTERS]
16 ;Junction Coefficient
17
18 [QUALITY]
19 ;Node InitQual
20 WIZ_13_10 20
21
22 [SOURCES]
23 ;Node Type Quality Pattern
24
25 [COORDINATES]
26 ;Node X-Coord Y-Coord
27 WIZ_13_10 641322.08 4817571.07
28

```

*Illustration 21: Struttura di un file di modello EPANET*

Poichè il sistema riceve un file contenente le coordinate dei nodi già presenti nel modello EPANET, l'utente dovrà anche indicare il sistema di riferimento utilizzato per quelle coordinate, in modo tale che, se necessario, il sistema converta le nuove coordinate generate nel sistema di riferimento voluto. In Figura 22 viene rappresentato il processo di generazione di un nuovo file EPANET nella sua interezza.



*Illustration 22: Generazione di un nuovo file EPANET*

## 6.8 Qualità Partecipata

Il sistema mette a disposizione, tipicamente per il cittadino, delle funzionalità per esprimere un giudizio di qualità (percepita) sull'acqua. Il giudizio si compone di una parte testuale, cioè la qualità espressa, e una parte geografica, cioè un punto sulla mappa nel quale il giudizio si riferisce.

Analogamente per la qualità, sono previste anche funzionalità per la segnalazione di guasti.

Il gestore del sistema può decidere quali sono i possibili livelli di qualità e di guasti. In fase di installazione il sistema prevede 5 livelli di qualità:

- eccellente
- buona
- decente
- bassa
- scarsa

e 2 livelli di guasti:

- perdita di acqua
- guasto generico

Scegliendo un livello di qualità o di guasto e un punto sulla mappa il sistema memorizza la coppia livello-punto georeferenziato e la rende disponibile per gli altri utenti.

## 6.9 Le informazioni geografiche

Il sistema memorizza le informazioni geografiche su un database Postgres con estensione PostGIS. Il server di mappe utilizzato è GeoServer.

Le informazioni geografiche che vengono visualizzate possono essere divise in due macrocategorie:

- dati di sfondo
- dati di overlay

I dati sfondo riguardano la rappresentazione del territorio; quelli previsti sono:

- OpenStreetMap: mappa fisica del territorio
- Ortofoto 10K: viene caricata sfruttando il servizio GEOscopio offerto dalla Regione Toscana
- CTR 10K e 2K: vengono caricate sfruttando il servizio GEOscopio offerto dalla Regione Toscana

I dati di overlay, invece, rappresentano delle informazioni che possono essere sovrapposte ai dati di sfondo (che invece sono mutuamente esclusivi). I dati di overlay attivabili dall'utente sono:

- Confini comunali: riporta, in formato shape, i limiti amministrativi dei comuni gestiti da Acque SPA.
- Aree di Servizio: riporta, in formato shape, la suddivisione dei comuni gestiti da Acque SPA in sottozone, denominate Aree di Servizio
- Geometrie: rappresentano le geometrie che compongono una data richiesta di risorsa idrica
- Rete di distribuzione: contiene informazioni sulla rete di distribuzione dell'acqua, georeferenziata, con le sue caratteristiche tecniche e di servizio
- Fonti: mostra l'esatta ubicazione delle fonti, suddivise in:
  - corsi d'acqua
  - laghi
  - pozzi
  - sorgenti
- Impianti: mostra l'esatta ubicazione degli impianti, suddivisi in:
  - potabilizzazione
  - adduttrici
  - accumulo
  - pompaggio
- Stradario: nomi di strade, piazze e luoghi, georeferenziati.

Per interrogare GeoServer e visualizzare le informazioni vengono utilizzare le funzionalità messe a disposizione dalla libreria OpenLayers.

## **6.10 Log & Feedback**

Oltre che per le notifiche, descritte nel capitolo "Le notifiche", il sistema invia delle email all'amministratore quando si verifica un malfunzionamento. In generale, il sistema riesce ad individuare automaticamente delle situazioni 'anomale', indipendentemente dal fatto che queste generino un errore visibile all'utente o meno; queste situazioni anomale vengono memorizzate in un file di log che riporta, oltre alla data e all'ora, una descrizione dettagliata sul malfunzionamento rilevato. Nei casi in cui l'errore è critico, e cioè causa anche un blocco del sistema, la descrizione dell'errore viene anche inviata per email all'amministratore del sistema; in questo modo l'amministratore può conoscere quali sono le situazioni critiche ed intervenire tempestivamente.

Esistono casi in cui il sistema non riesce ad accorgersi in automatico della presenza di un guasto; e questo vale soprattutto per le interfacce utente, 'sensibili' ai vari browser e alle varie versioni. Per questo motivo è stato sviluppato un meccanismo di invio feedback semplice da utilizzare ma potente per determinare il tipo di guasto. Tramite un link posto in basso a destra dello schermo, e sempre visibile da qualsiasi tipologia d'utente, è possibile catturare la schermata, aggiungendo eventualmente delle note o oscurando dei dati sensibili; il sistema invierà l'immagine della schermata all'amministratore, aggiungendo anche informazioni sul sistema operativo dell'utente che ha generato il feedback, ed il tipo e la versione di browser utilizzato.

## **6.11 Internazionalizzazione**

Il portale include funzionalità per l'internazionalizzazione. Le lingue disponibili sono l'inglese e l'italiano; chiaramente è possibile aggiungere una qualsiasi lingua fornendo preventivamente i file delle traduzioni.

La lingua non può essere scelta dall'utente finale ma viene settata in fase di installazione del sistema. L'utente amministratore può modificarla in qualsiasi momento; questa modifica interesserà tutti gli utenti che vedranno il portale nella lingua aggiornata.

I file delle traduzioni risiedono nella cartella `messages/<language_code>/`; ogni file stabilisce una corrispondenza chiave-traduzione nella quale la chiave viene utilizzata, all'interno del codice, per visualizzare la traduzione.

Per creare, quindi, una nuova traduzione basta replicare i file presenti dentro la cartella `messages/<language_code>/` ed effettuare la traduzione sono della parte 'traduzione', lasciando inalterata la parte 'chiave'. Questi file andranno memorizzati all'interno della cartella `messages/<new_language_code>/`; per abilitare la traduzione, la nuova lingua deve essere impostata tra i parametri di configurazione. Per maggiori informazioni sui parametri si rimanda al capitolo .

## **6.12 Configurazione parametri**

Tutti i parametri di configurazione del portale risiedono nel file `config/params.php`. Questi parametri permettono di specificare il comportamento di alcune componenti del sistema e di modificarlo facilmente secondo le necessità. Il sistema prevede anche una semplice interfaccia utente tramite la quale l'amministratore del sistema può modificare il valore di questi parametri o eventualmente aggiungerne altri.

La lista seguente elenca tutti i parametri utilizzati e li descrive brevemente:

- `language`: lingua del portale. Valore di default: 'it'

- adminEmail: indirizzo email dell'amministratore al quale inviare comunicazioni sul funzionamento generale del sistema. Valore di default: `acque@cpr.it`
- debugEmail: indirizzo email dell'amministratore al quale inviare comunicazioni sugli errori che si verificano durante l'utilizzo del sistema. Valore di default: `acque@cpr.it`
- block\_debugEmail: indica se bloccare l'invio delle mail di debug. Valore di default: `false`
- block\_email: indica se bloccare l'invio di tutte le mail generate dal sistema. Valore di default: `false`
- geoserver: parametro composto. Contiene informazioni relative a geoserver:
  - version: version di geoserver installata. Valore di default: `1.1.0`
  - protocollo: protocollo di comunicazione con geoserver. Valore di default: `http://`
  - ip: indirizzo ip di geoserver. Valore di default: `127.0.0.1`
  - porta: porta di ascolto di geoserver. Valore di default: `8080`
  - path: path in cui inviare le richieste per geoserver. Valore di default: `/geoserver`
  - wms: path per accedere ai servizi wms di geoserver. Valore di default: `/wms`
  - wfs: path per accedere ai servizi wfs di geoserver. Valore di default: `/wfs`
  - workspace: workspace all'interno del quale sono memorizzati i dati. Valore di default: `acque`
  - layer\_geom: nome del layer che memorizza le geometrie. Valore di default: `wr\_geom`
  - default\_srs: sistema spaziale di riferimento. Valore di default: `EPSG:4326`
  - citystate\_layer\_srid: sistema spaziale di riferimento utilizzato per i confini comunali. Valore di default: `3003`
  - water\_request\_geometries\_srid: sistema spaziale di riferimento utilizzato per le geometrie di una water request. Valore di default: `4326`
  - pdf\_geoms: nome del layer delle geometrie utilizzato per la generazione dei pdf. Valore di default: `pdf\_geoms`
  - layer\_dem: nome del layer DEM. Valore di default: `DEM`

- `service_areas_layer_srid`: sistema spaziale di riferimento utilizzato per le aree di servizio. Valore di default: `'3003'`
- `water_demand_unit`: unità di misura dell'idroesigenza. Valore di default: `'l/s'`
- `decimals`: numero di cifre dopo la virgola. Valore di default: `'2'`
- `pe_precision`: numero di cifre dopo la virgola per gli abitanti equivalenti. Valore di default: `'0'`
- `we_decimals`: numero di cifre dopo la virgola per l'idroesigenza. Valore di default: `'2'`
- `pe_abbreviation`: sigla indicante gli abitanti equivalenti. Valore di default: `'PE'`
- `formula`: parametro composto. Contiene informazioni relative alle formule per il calcolo dell'idroesigenza:
  - `pe`: sigla utilizzata per indicare gli abitanti equivalenti. Valore di default: `'ae'`
  - `da`: sigla utilizzata per indicare la dotazione media annua. Valore di default: `'da'`
  - `dg`: sigla utilizzata per indicare la dotazione massima giornaliera. Valore di default: `'dg'`
- `dateFormat`: indica come formattare i campi datetime per rendere la data e l'ora leggibile all'utente. Valore di default: `'j F Y, H:i'`
- `datetimeFormat`: indica come formattare i campi datetime per poter essere memorizzati correttamente sul database. Valore di default: `'Y-m-d H:i:sO'`
- `dateFormat`: indica come formattare i campi date per rendere la data leggibile all'utente. Valore di default: `'j F Y'`
- `EPANET`: parametro composto. Contiene informazioni relative all'interfaccia per EPANET:
  - `section_marker_start`: carattere indicante l'inizio del tag sezione, all'interno di un file generato da EPANET. Valore di default: `'['`
  - `section_marker_end`: carattere indicante la fine del tag sezione, all'interno di un file generato da EPANET. Valore di default: `']'`
  - `section_regular_expression`: espressione regolare utilizzata per identificare il nome di una sezione. Valore di default: `'/^\\[[a-zA-Z0-9]{1,}\\]'`

- `comment_marker`: carattere indicante l'inizio di un commento, all'interno di un file generato da EPANET. Valore di default: `;`
- `section_junctions`: nome della sezione junctions, all'interno di un file generato da EPANET. Valore di default: `JUNCTIONS`
- `section_coordinates`: nome della sezione coordinates, all'interno di un file generato da EPANET. Valore di default: `COORDINATES`
- `section_tags`: nome della sezione tags, all'interno di un file generato da EPANET. Valore di default: `TAGS`
- `section_emitters`: nome della sezione emitters, all'interno di un file generato da EPANET. Valore di default: `EMITTERS`
- `section_quality`: nome della sezione quality, all'interno di un file generato da EPANET. Valore di default: `QUALITY`
- `section_sources`: nome della sezione sources, all'interno di un file generato da EPANET. Valore di default: `SOURCES`
- `tag_node_junctions`: valore del campo node all'interno di tag, in un file generato da EPANET. Valore di default: `NODE`
- `source_type_junctions`: valore del campo type all'interno di source, in un file generato da EPANET. Valore di default: `CONCEN`
- `upload_dir`: indica la directory nella quale memorizzare i file EPANET che vengono caricati dall'utente. Valore di default: `epanet\_uploads`
- `upload_max_size`: indica la dimensione (in MegaByte) massima di un file EPANET. Valore di default: `10`
- `download_dir`: indica la directory nella quale memorizzare i file EPANET generati dal sistema e che dovranno essere scaricati dall'utente. Valore di default: `epanet\_download`
- `download_filename_suffix`: rappresenta il suffisso che viene inserito nel nome del file prima che venga scaricato dall'utente. Valore di default: `-WIZ-GENERATED`
- `junction_prefix`: rappresenta il prefisso che viene preposto al nome delle junctions inserite dal sistema all'interno di un file EPANET. Valore di default: `WIZ\_`
- `src_download_folder`: indica la directory che contiene l'archivio del codice sorgente del portale. Valore di default: `/download/`
- `src_zip_file`: indica il nome del file dell'archivio in formato zip del codice sorgente del portale. Valore di default: `wiz\_latest.zip`

- `src_tar_file`: indica il nome del file dell'archivio in formato zip del codice sorgente del portale. Valore di default: `'wiz_latest.tar'`
- `shp_upload_folder`: indica la directory nella quale vengono memorizzati gli shape caricati dagli utenti. Valore di default: `'../uploads/shp/'`
- `transition`: parametro composto. Contiene informazioni riguardanti il passaggio di stato di una water request:
  - `upload_dir`: indica la directory nella quale memorizzare l'archivio da allegare ad una water request dopo l'approvazione tecnica. Valore di default: `'uploads/wrut'`
  - `upload_max_size`: indica la dimensione (in MegaByte) massima dell'archivio. Valore di default: `'10'`
- `plugins`: parametro composto. Contiene informazioni riguardanti i plugins:
  - `upload_dir`: indica la directory nella quale memorizzare i plugin che vengono caricati. Valore di default: `'plugins'`
  - `upload_max_size`: indica la dimensione (in MegaByte) massima del plugin. Valore di default: `'20'`

## 7 Glossario

PE = AE = Abitanti equivalenti

ds = dg = dotazione comunale giornaliera massima

ys = da = dotazione comunale annuale media

sul = superficie utile lorda

ui = unità immobiliari

sup = superficie territoriale

UTOE = Unità Territoriali Organiche Elementari

## **8 Appendice A: Schema del Database**

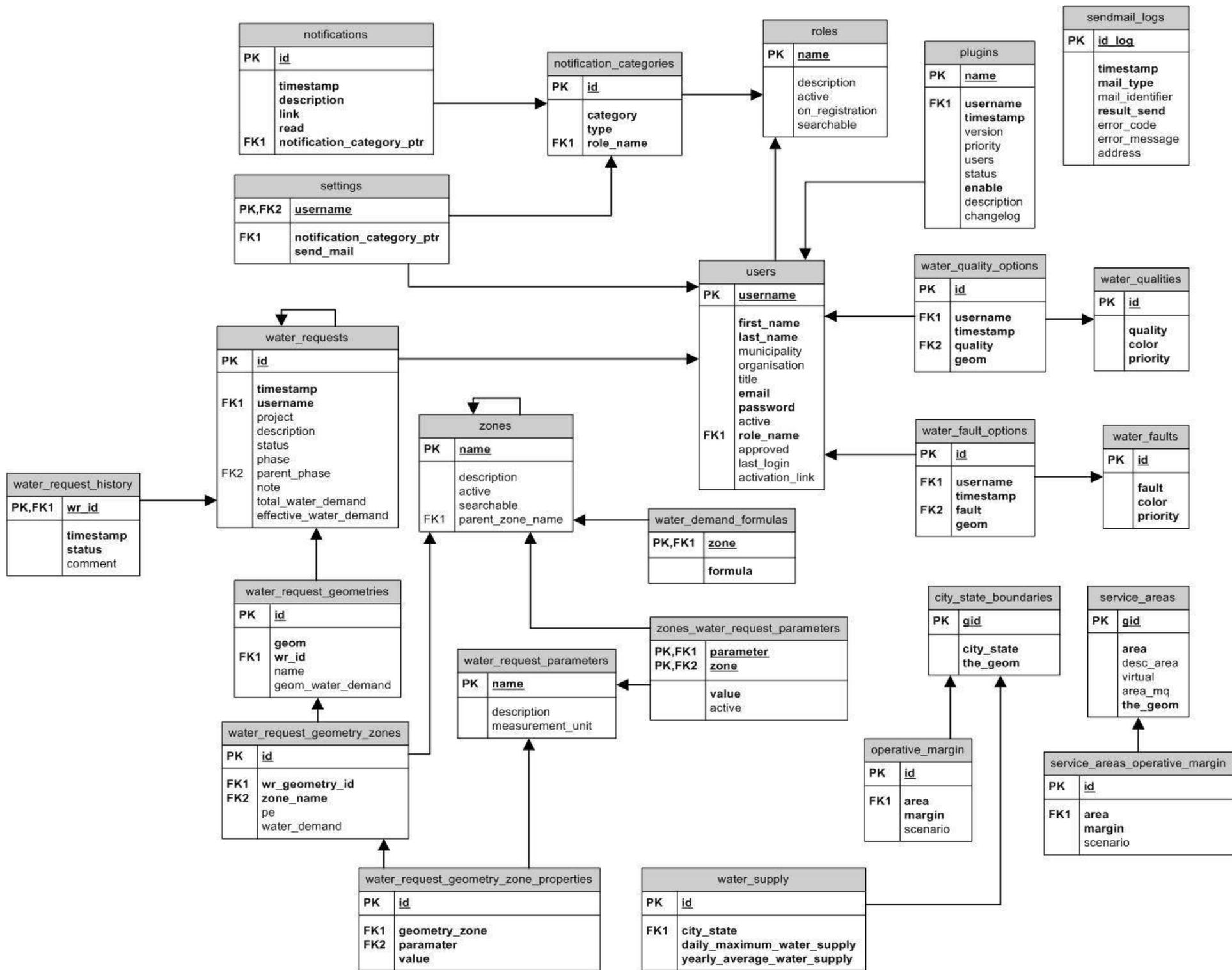


Illustration 23: Schema database