



IUT de VANNES
Département Informatique
BP 561
56017 VANNES CEDEX

REUX Matthias
Groupe C
MERCIOLO François



Stage de conception de base de données météorologiques

Stage effectué du 06 avril au 12 juin 2015 en vue de l'obtention de DUT Informatique, sous la responsabilité de :

THOMAS Alban, Ingénieur d'étude cartographe
QUÉNOL Hervé, Directeur de recherche

Au sein de l'entreprise :

UMR LETG Rennes COSTEL
Université Rennes 2
Place du Recteur Henri Le Moal
35043 RENNES CEDEX



IUT de VANNES
Département Informatique
BP 561
56017 VANNES CEDEX

REUX Matthias
Groupe C
MERCIOLO François

Stage de conception de base de données météorologiques

Stage effectué du 06 avril au 12 juin 2015 en vue de l'obtention de DUT Informatique, sous la responsabilité de :

THOMAS Alban, Ingénieur d'étude cartographe
QUÉNOL Hervé, Directeur de recherche

Au sein de l'entreprise :

UMR LETG Rennes COSTEL
Université Rennes 2
Place du Recteur Henri Le Moal
35043 RENNES CEDEX

Remerciements

Je tiens à remercier l'ensemble du Costel pour m'avoir accueilli durant la durée de ce stage, et tout particulièrement Alban THOMAS, mon maître de stage, qui s'est donné la peine de suivre régulièrement l'avancée de mon travail et de m'éclairer sur des points qui m'étaient obscurs. Je tiens également à remercier Nicolas OLIVIER, mon collègue stagiaire avec qui j'ai collaborer sur quelques courtes parties du stage et sur des questions de logistique (organisation de la visite du lieu de stage prévue). Je remercie aussi l'IUT de Vannes et son équipe enseignante qui m'ont donné l'opportunité d'effectuer ce stage et les moyens de le mener à bien.

Sommaire

Remerciements.....	4
I Présentation de l'entreprise.....	6
I 1 LETG Rennes Costel.....	6
I 2 la Zone Atelier Armorique (ZAA).....	7
I 3 l'Observatoire des Sciences de l'Univers de Rennes (OSUR).....	8
II Contexte du stage.....	9
II 1 Présentation de l'existant.....	9
II 2 Problématique.....	11
II 3 Solution à apporter.....	12
III Travail effectué.....	14
III 1 Phase préparatoire.....	14
III 2 Phase de modélisation.....	14
III 2 1 L'interface de consultation.....	14
III 2 2 La base de données.....	15
III 3 Création de la base.....	19
III 4 Les scripts d'import.....	20
III 5 L'interface de consultation.....	22
III 6 Documentation.....	22
IV Bilan.....	23
IV 1 Objectifs remplis.....	23
IV 2 Possibilités d'évolution.....	23
IV 3 Formation technique acquise.....	23
IV 4 Bilan personnel.....	23
Bibliographie.....	24

I Présentation de l'entreprise

Afin de clôturer ma formation à l'IUT, j'ai effectué un stage d'une durée de dix semaines à l'UMR LETG Rennes Costel (Climat, Occupation du Sol et Télédétection), l'un des 4 sites de l'UMR LETG (Littoral, Environnement, Télédétection et Géomatique) dont les locaux sont situés à l'université de Rennes 2. Lors de ce stage j'ai pris part à un projet qui s'intègre dans le cadre de l'Observatoire des sciences de l'univers de Rennes (OSUR) et porté sur les sites d'études de Zone Atelier Armorique (ZAA). Je détaillerai ces trois entités ci-après.

I 1 LETG Rennes Costel

LETG est une UMR (Unité mixte de recherche), composée de personnels des universités et du CNRS. Ses activités de recherche s'inscrivent dans les champs de la géographie et de l'environnement, comme le souligne l'acronyme LETG (pour Littoral, Environnement, Télédétection et Géomatique). Les recherches du laboratoire privilégient l'approche spatiale des interactions entre nature et société en mettant l'accent sur l'étude des processus dynamiques naturels et anthropiques qui contribuent à définir les milieux et les paysages, les changements globaux et les risques climatiques.

LETG est une unité multi-site, implantée dans cinq universités de l'Ouest de la France :

- l'Université de Bretagne Occidentale (UBO, Brest)
- l'Université de Caen-Basse-Normandie (UCBN)
- l'Université de Nantes
- l'Université de Rennes 2
- l'Université d'Angers

Le site qui m'accueille pour mon stage, Costel, est celui situé sur Rennes et c'est tout naturellement qu'il a intégré l'OSUR et qu'il s'intéresse aux sites de la ZAA.

Les recherches menées par Costel s'inscrivent notamment dans les domaines de la climatologie, la télédétection et de l'écologie.

I 2 la Zone Atelier Armorique (ZAA)

Les Zones ateliers sont un dispositif interdisciplinaire, inter-instituts et inter-organismes de recherche à long terme sur l'environnement et les interactions hommes-milieu portés par le CNRS (Centre National de la recherche scientifique) depuis 14ans. Elles constituent un service national d'observation et de recherche en environnement.

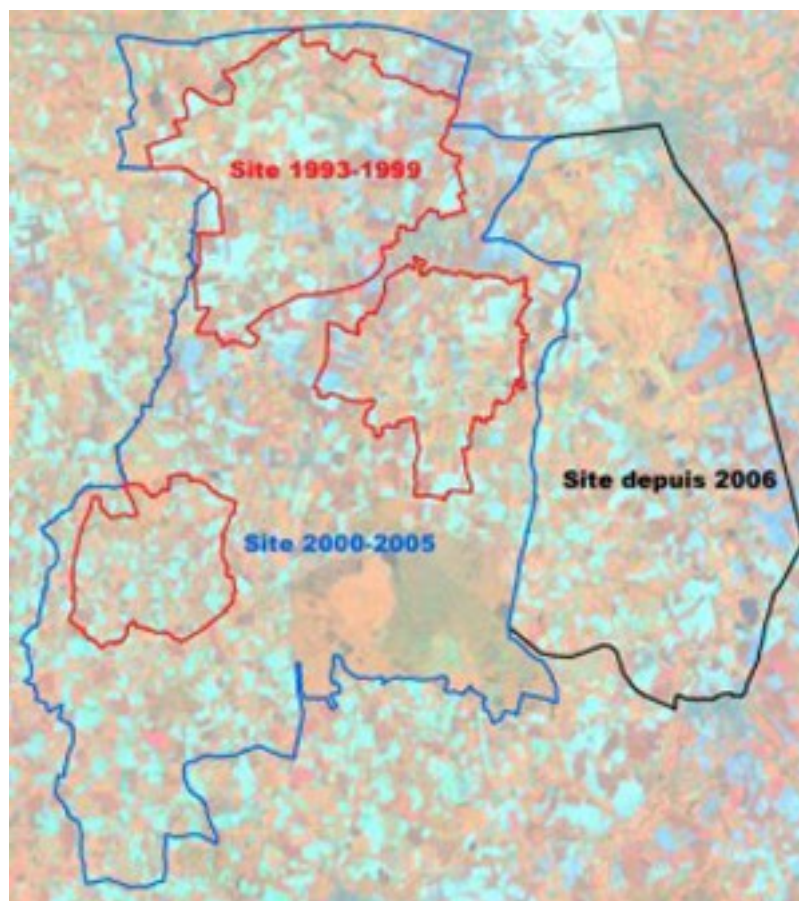


Figure 1: Limites de la ZAA

La ZA Armorique mène des recherches sur les relations entre dynamiques sociétales et dynamiques environnementales, le long de gradients allant de l'urbain au rural, avec le paysage comme objet d'articulation. Les sites d'études de la ZA Armorique se trouvent sur Rennes (milieu urbain) et au nord de Rennes (milieu rural) afin de pouvoir observer l'évolution des données suivant l'urbanisation du milieu .

Tant qu'ils respectent les règles et objectifs de la ZAA, plusieurs acteurs peuvent y effectuer des recherches de nature différentes et, éventuellement, mutualiser des données et des équipements. Par exemple sur le site de Rennes, Rennes Métropole a contribué aux recherches effectuées dans cette zone en fournissant des stations météorologiques.

I 3 l'Observatoire des Sciences de l'Univers de Rennes (OSUR)

Les observatoires des sciences de l'univers (OSU) sont des organismes, créés par les ministères de la recherche et de l'enseignement supérieur, gérés par l'INSU (Institut National des Sciences de l'Univers, branche du CNRS) dont la mission est de contribuer aux progrès de la connaissance par l'acquisition de données d'observation, le développement et l'exploitation de moyens appropriés et l'élaboration des outils théoriques nécessaires. Ils sont également chargés de fournir des services liés à leur activités de recherche, d'assurer la formation des étudiants et des personnels de recherche, d'assurer la diffusion des connaissances (auprès de l'enseignement) et des activités de coopération internationale

L'OSUR (Observatoire des Sciences de l'Univers de Rennes) est un OSU lié à l'université de Rennes chargé d'organiser l'enseignement et la recherche dans le domaine des sciences de la Terre, de l'écologie et de l'environnement dans la zone de Rennes. L'OSUR est constitué de 4 unités ou équipes de recherche, dont LETG Rennes Costel, et d'une UMS (unité mixte de service) et s'associe, par convention de partenariat, à diverses unités ou équipes de différents domaines scientifiques. La majeure partie des acteurs de la ZAA sont liés à l'OSUR et une part importante des contributeurs de l'OSUR est concernée par les recherches effectuées sur la ZAA, ce sont cependant deux entités distinctes.

Les données que j'ai manipulées durant ce stage doivent être mises à disposition des membres de l'OSUR immédiatement, et pourra être éventuellement accessible, partiellement ou complètement, au grand public. Cette mise à disposition résulte d'une volonté du Costel et s'inscrit dans le cadre de la directive Européenne Inspire.

II Contexte du stage

II 1 Présentation de l'existant

LETG Rennes Costel gère un réseau de stations météorologiques dispersées au travers de la ZAA (18 stations en fonction sur Rennes et 6 sur Pleine-Fougères), dont les premières données datent de 1996. Ces stations météorologiques sont de plusieurs modèles différents :

- Tinytag : Capteur de température



- Davis Weather Monitor 2 et Vantage Pro2 : stations météorologiques modulables capables de mesurer plusieurs données (Température, humidité, pression ...)



← Weather Monitor 2



Vantage Pro 2 →

- Campbell CR200 : station météorologique capables de de mesurer plusieurs données, comme les Davis, mais plus fiable et dotée d'un module de communication GSM



Ces modèles de station n'acquièrent pas toutes les mêmes données : uniquement la température pour les Tinytags, alors que les autres modèles acquièrent en plus la pression atmosphérique, les précipitations, la vitesse et la direction du vent. En fait, le nombre de données est lié aux capteurs connectés aux stations et ce nombre est variable. Par exemple, plusieurs capteurs de température peuvent être connectés à une station. Ainsi, chaque modèle de station livre ces données selon un format qui lui est propre (mais toujours disponible sous forme texte structuré), et un même modèle de station peut fournir

différents formats de données. De plus, les stations demandent un mode de relevé des données différent.

Actuellement un opérateur (Alban Thomas) relève régulièrement les données de 11 stations sur Rennes et des six stations sur Pleine-Fougère. Les 7 autres stations sur Rennes sont gérées par des volontaires qui envoient les données par mail. La volonté de l'UMR est de réduire le temps nécessaire à la récolte de ces données et une automatisation de la récolte des données est en cours de développement :

- d'une part, avec des systèmes autonomes (Raspberry) qui collectent et envoient les données
- d'autre part, par communication GSM pour les station Campbell

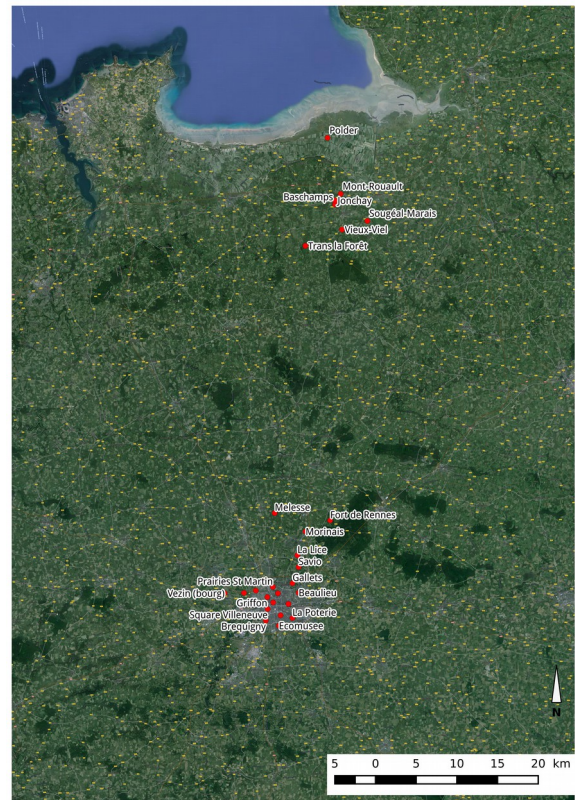


Figure 2: Localisation des stations météorologiques

Au delà de l'acquisition des données, LETG a une volonté de faire connaître les données qu'elle produit et s'est doté d'une Infrastructure de Données Géographique nommée Indigeo. Naturellement, les données météorologique de la ZAA ont vocation à y figurer dans le futur mais dépasse le cadre de ce stage. La constitution d'une base de données et d'une interface de consultation sont deux préalables nécessaires avant d'envisager une intégration dans Indigeo.

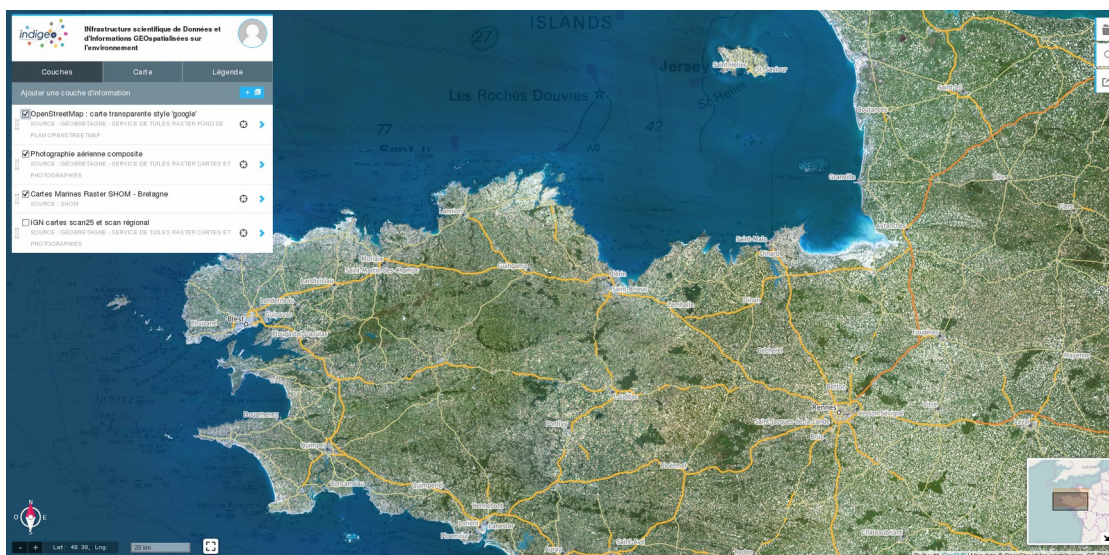


Figure 3: Interface d'Inigéo

II 2 Problématique

L'ensemble des données météorologiques est stocké sous forme de fichiers texte formatés de différentes manières. Le nombre de colonnes, correspondant au nombre de paramètres météorologiques relevés, peut varier suivant le modèle de station et les capteurs installés, la forme de l'entête peut varier suivant le modèle de station.

Les données ne sont pas centralisées dans un système d'information, rendant laborieuse toute recherche d'informations fiables. Sans possibilité de vérification automatique, les données peuvent contenir des erreurs liées notamment à des défaillances des capteurs (ex : un capteur de pluviométrie qui se bouche, entraînant une sécheresse localisée) où à une erreur de l'opérateur.

De plus la consultation de ces données demande qu'un opérateur fouille dans l'ensemble de fichiers texte afin de trouver l'information souhaitée. Ce temps pourrait être mieux utilisé.

La gestion du parc de stations se fait également par l'intermédiaire d'un fichier tableur, ce qui rend difficile à maintenir et à lire l'état global du parc et les interventions faites / restant à faire.

II 3 Solution à apporter

Les objectifs qui m'ont été assignés sont de :

- Mettre en place une base de données permettant de stocker les données provenant de tout type de stations météorologique, ainsi que des informations concernant le réseau de stations météorologique (type de capteurs sur chaque station, maintenance réalisée, erreur potentielles, ...). La base de donnée doit également pouvoir intégrer aisément tout nouveau format de fichier de données sans devoir subir un remaniement complet.
- Créer un outil capable d'interpréter les fichiers de données afin d'insérer dans la base les données contenus dans ces fichiers sources après un premier contrôle de la qualité des données. Les méthodes d'insertion doivent conserver l'intégrité des données de la base et fichiers sources ; en cas de conflit, l'opérateur devra choisir d'altérer une des données ou d'annuler l'import.
- Permettre à un utilisateur de consulter de manière simple les données recueillies par l'intermédiaire d'une interface dédiée. L'utilisateur devra pouvoir signaler des valeurs douteuses qui seront retirée de la consultation sans pour autant altérer l'intégrité de la base de donnée. Cette consultation ne se fera qu'en intranet dans un premier temps mais la solution pourrait être portée sur Indigeo.
- Permettre aux administrateurs du parc de stations de visualiser rapidement les problèmes signalés sur les stations et de gérer plus efficacement la récolte des données grâce à une interface administrateur dédiée.

Le cahier des charges défini en début de stage est disponible ci-après.

Cahier des charges

-Conception de la structure de la base de données(PgModeler)

Une base de données "source" (entrepôt) + 1 base de données de traitement

Base de traitement alimentée par l'entrepôt contenant :

- les données captées
- les informations portant sur les sites/stations/capteurs
- les données des interventions/incidents sur les stations

- Données

Import de données depuis différents types de stations :

- Campbell(GSM)
- stations Davis via Weatherlink ou Raspberry
- TinyTags
- Stations de référence (MeteoFrance)
- + autres (possibilité d'adaptation)

Chargement des données via un script Python

Reconnaissance du type de station par le script

Vérification de la qualité des données :

- Vérification des unités (Système International)
- Vérification date/heure (Heure UTC)
- Vérification données aberrantes
- Comparaison avec les données de référence

- Consultation de la base de donnée

Mise en place d'une interface(intranet dans un premier temps)

- interface ergonomique, simple, rapide, intuitive
- toutes les stations et paramètres sélectionnés par défaut
- modulation de la plage temporelle (période totale et intervalle de mesure)
- possibilité d'exporter des résultats dans plusieurs formats : csv, excel
- possibilité de signaler des données suspectes

Utilisation de la base de traitement pour l'interaction

Interface administrateur avec état des stations(révisions, problèmes signalés,...)

-Rendus attendus

- Bilans hebdomadaires
- Rapport de stage
- Manuel administrateur

-Outils à utiliser

- Base de données PostGres + PostGis
- Outil de modélisation de base de données PgModeler
- Python
- Ruby (potentiellement)
- BitBucket(git)
- Gantt

III Travail effectué

III 1 Phase préparatoire

Après une visite des locaux et une mise à disposition d'un poste équipé des outils nécessaires à stage, nous entamons une réunion afin de discuter et de redéfinir clairement les objectifs du stage (décrits ci-dessus).

Le travail demandé lors de mon stage se compose de plusieurs tâches, certaines où je serai autonome et d'autres en collaboration. J'ai ainsi réalisé ce stage aux côtés de :

- Alban THOMAS : mon maître de stage et Ingénieur d'étude cartographe
- Nicolas OLIVIER : autre stagiaire issu de l'IUT de Vannes réalisant un stage concernant également des données météorologiques dont le sujet est : "Lien entre une base de données PostgreSQL et un algorithme R"
- Renan LeRoux : maître de stage de Nicolas OLIVIER et doctorant

III 2 Phase de modélisation

Cette phase de modélisation, conduite en coopération avec Nicolas, a permis d'élaborer 2 modèles : un modèle de base de donnée et un modèle pour l'interface de consultation.

III 2 1 L'interface de consultation

Pour l'interface, nous avons utilisé une simple feuille de papier et un crayon afin de réaliser une esquisse du visuel souhaité et lister les réactions attendues. Nous sommes partis sur l'idée d'une interface visuellement simple et épurée, comme demandé dans la cahier des charges, en noir sur fond blanc, dont les démarcations entre les sections relèvent plus du placement réfléchi des éléments que de bordures visibles. Nous souhaitons une interface dynamique ne nécessitant qu'une unique page de consultation dans l'esprit "Single Page Application", en plus d'une page d'administration réfléchi selon les mêmes critères. L'utilisateur devrait pouvoir modifier des paramètres et avoir un retour instantané (ou perçu comme tel) dans la zone de visualisation des données, nous avons donc réfléchi à une organisation de l'espace permettant de faire tenir sur un seul écran le panneau de paramètres et un maximum de données en les séparant de manière claire. Une fonctionnalité permettant d'exporter les données sélectionnées était également souhaitée.

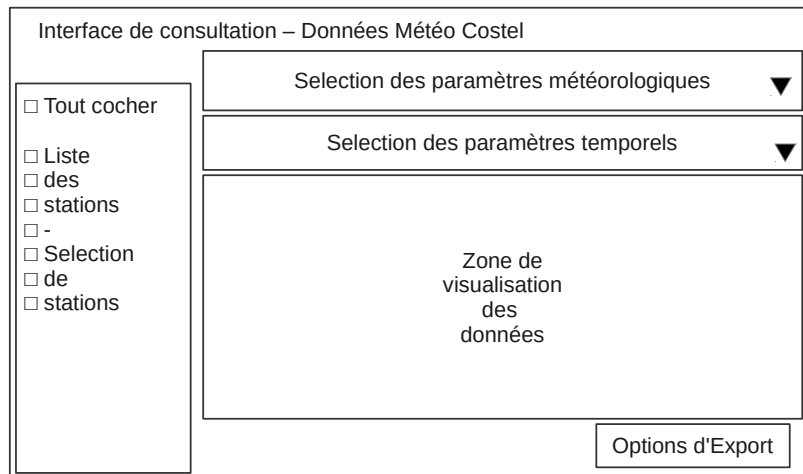


Figure 4: Modèle de l'interface de consultation

III 2 2 La base de données

Pour la base de données, nous utilisons une base de données PostgreSQL (v 9.3) avec son module géographique (PostGis), imposé par la nécessité de pouvoir manipuler des objets géographiques (les sites des stations météo). De plus, la plate-forme Indigeo utilisant déjà une base de données PostgreSQL avec l'extension PostGis, cela facilitera l'importation de la base de données ainsi conçue vers cette plateforme. Pour cette phase de modélisation nous avons utilisé l'outil graphique PGModeler (v 0.8) qui permet de générer automatiquement du code SQL adapté à une base PostgreSQL. L'outil inclut également une fonction de rétro-conception efficace, ce qui m'a permis, combiné à la génération de code, d'inclure dans le modèle des éléments difficiles à modéliser à partir de l'interface de PGModeler (le code des fonctions servant au contrôle des données et à la définition de la vue matérialisée principalement).

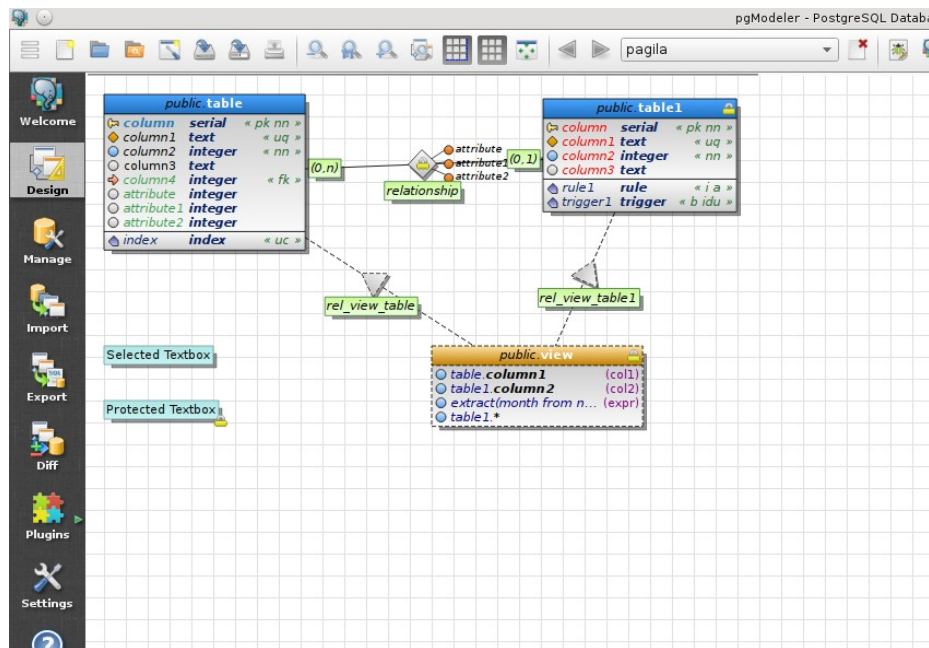


Figure 5: Interface de l'outil PgModeler

Les fichiers de données vont être découpés ligne par ligne et chaque ligne de données va devenir un tuple d'une table. Pour pallier le problème des différents formats de données possibles nous utiliserons une particularité de PostgreSQL : c'est un SGBD relationnel et objet, ce qui nous permet d'utiliser la propriété d'héritage pour faire autant de tables que de formats de données mais en mutualisant certaines informations, et contraintes, communes à toutes les lignes de données (identifiant, date, relevé associé, identifiant au sein du relevé) afin de simplifier la structure de la base et le traitement de ces données. Le relevé représente quand à lui un ensemble de lignes de données relevées sur une station à une heure donnée par une personne (ou un système autonome) donnée. L'ensemble de ces données est stockée dans la base au plus proche des données collectées et ne doivent pas être altérées par la suite (sauf cas spéciaux gérés par l'administrateur de la base). Ces données font donc partie d'un schéma dont l'accès est restreint aux seuls administrateurs de la base afin d'en assurer l'intégrité.

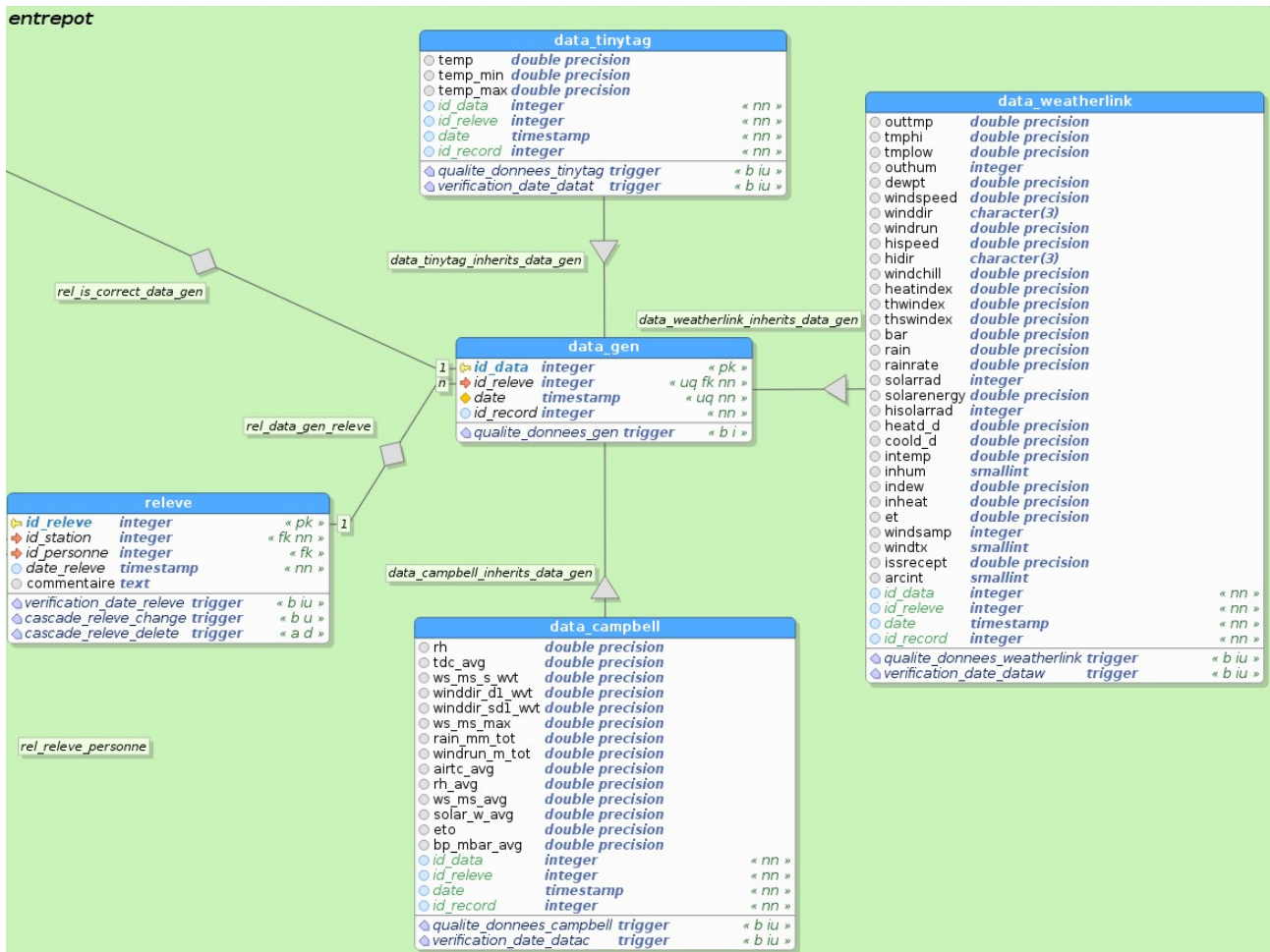


Figure 6: modèle de base de données, partie entrepot de données

Les données provenant de ces tables servent ensuite à alimenter une vue matérialisée (stockée dans le schéma public avec le reste des données) permettant de rassembler toutes les données sous un seul format dans une seule relation afin de faciliter la lecture. Nous aurions pu formater les données dès leur insertion dans la base mais nous souhaitons conserver une structure de base aussi proche que possible de la structure des fichiers sources.

Une vue matérialisée est table virtuelle n'étant pas rafraîchi automatiquement qui représente le résultat d'une requête sur la base lors de sa dernière exécution. Cela permet d'éviter la duplication des données sources et prévient de la modification de ces dernières en cas de modification de la vue.

La base de données doit également contenir des informations concernant le matériel utilisé (stations et capteurs), les interventions d'opérateur sur le réseau de stations et des données permettant de vérifier ces informations (et : compatibilité entre les modèles de station et les modèles de capteurs) et de les relier entre elles (ex : table d'utilisation d'un capteur par une station).

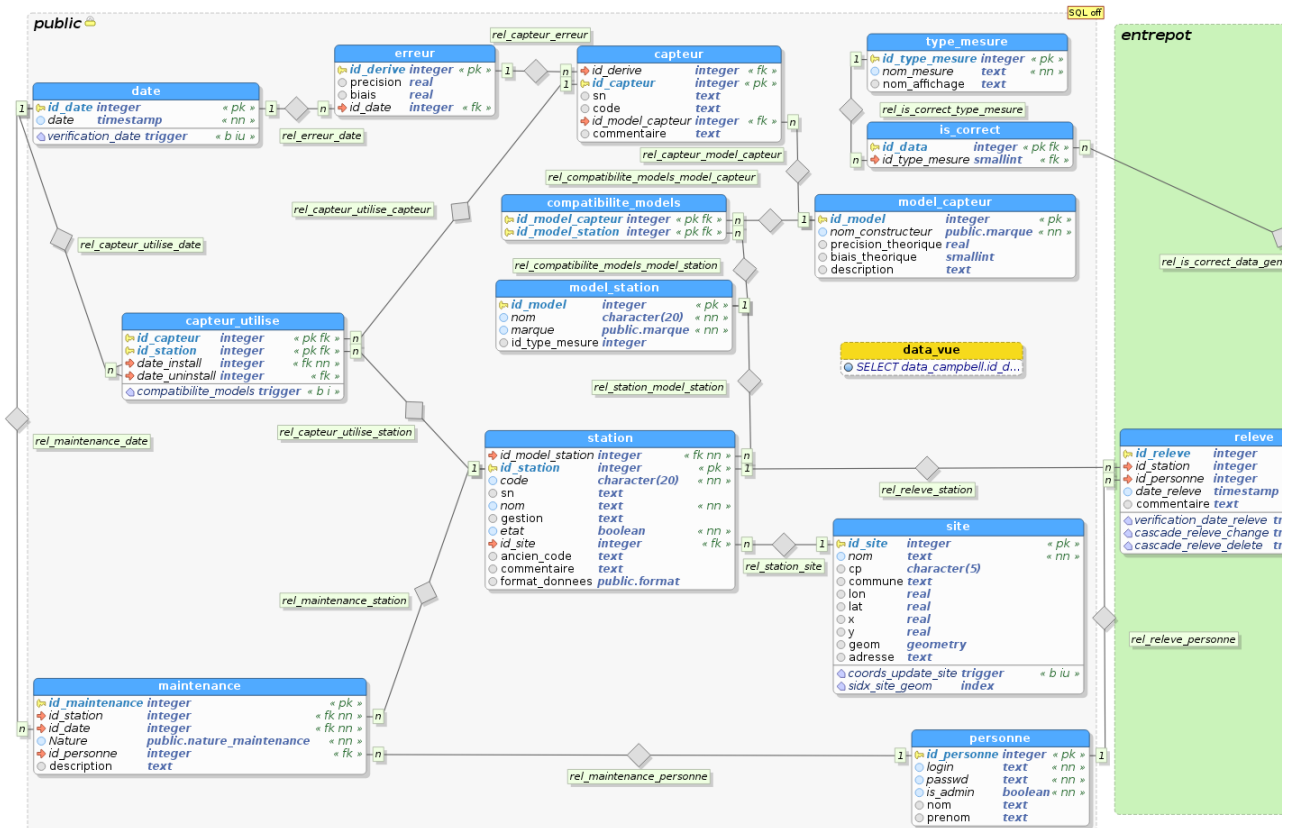


Figure 7: Modèle de base de données, partie publique

Le stage de Nicolas ne nécessitant pas une base de données aussi complète que celle-ci mais devant être fonctionnel plus vite, il travailla majoritairement sur le modèle d'interface tandis que j'améliorais le modèle de base de données.

III 3 Création de la base

La base de donnée ayant servi de base de test à été hébergée en local sur mon poste de travail.

Les premières base de données de test ont été mises en place avant la validation du modèle de base de donnée afin de réaliser quelques test avec PostgreSQL, que je découvrais tout juste, et avec le concept de vue matérialisée, qui n'aura été adoptée que tardivement.

Grâce à l'outil de génération de code SQL de PGModeler, une part importante du script de création de la base de données été achevée avant de passer à la phase de création, d'autant plus que certaines fonctions ont été développées durant la modélisation afin d'être insérer dans le modèle grâce à l'outil de rétro-conception, ces deux phases du stage étaient fortement liées.

Mais il se trouve que PGModeler à des soucis pour générer les vues matérialisées (et pour les rétro-concevoir également), il m'a donc fallu rédiger la définition de la vue matérialisée à la main en ajoutant une fonction de formatage pour convertir la direction du vent en code 3 de lettres depuis une valeur angulaire.

Comme précisé plus haut, PostgreSQL permet de créer des liens d'héritage entre table. Cependant les contraintes, sauf celle de non-nullité, ne sont pas héritées (voir les champs en vert sur la figure 6). Cela pose des problèmes notamment au niveau de la contrainte de clé primaire, en effet avec ce système le contenu de la table mère est l'union du contenu des table filles restreintes sur les colonnes héritées. La colonne portant la contrainte de clé primaire est donc héritée sans sa contrainte d'unicité et il est alors possible d'insérer dans la table fille deux lignes avec une valeur de clé identiques ; quand on fera une sélection sur la table mère on retrouvera là aussi deux ligne avec une valeur de clé identiques malgré la contrainte d'unicité. Pour pallier ce soucis j'ai mis en place des triggers réagissant à l'insertion de données pour simuler la présence de ces clés.

Les triggers ("déclencheur" en anglais) sont des événements déclenchés lorsqu'une action (insertion, suppression ou modification) prédéfinie est réalisée sur une table. Dans PostgreSQL, les triggers ne peuvent qu'appeler une qu'une "fonction de trigger" qui réalisera le traitement souhaité.

Les problèmes liés à l'héritage et aux vues matérialisées peuvent s'expliquer par le fait que ce sont des notions introduites relativement récemment dans PostgreSQL. En effet PostgreSQL est dans sa version 9.4, l'héritage à été introduit avec la version 9.0 et les vues matérialisées datent de la version 9.3. On peut alors comprendre que PGModeler ne traite pas encore parfaitement ces notions.

Toujours grâce à des triggers, j'ai également mis en place un système de contrôle de cohérence des dates (les opérations ne peuvent être effectuées dans le futur, et les lignes de données traitent de données antérieures à leur relevé) et de prévention d'écrasement de données qui vérifie si des données existent déjà pour le couple station/date donné et, le cas échéant, quels sont les différences entre les deux jeux de données.

III 4 Les scripts d'import

Les scripts d'import ont été réalisés avec l'aide d'Alban Thomas. Nous avons retenu le langage python car c'est le langage qu'Alban maîtrise le mieux et qu'il possédait des premières versions de scripts. Après une courte formation (1 journée) dispensée par Alban sur les bases du langage, je me suis auto-formé sur des concepts un peu plus avancés et sur l'utilisation des bibliothèques Pandas (permettant de créer et manipuler aisément différents types de structures de données) et Psycopg2 (permettant la communication avec une base de données).

Le projet à été mené avec une approche orientée objet utilisant des classes génériques afin d'ouvrir des possibilités de réutilisation du code. Les méthodes de ces classes, génériques comme plus spécifiques, font l'objet d'un jeu de tests unitaires (plus d'une vingtaine). Seul la séquence principale sera écrite de manière procédurale, bien qu'instanciant les classes du reste du modèle objet (voir figure 8 ci-après). Après avoir collaboré au modèle-objet, ma contribution au développement du code s'est focalisé sur les méthodes concernant la communication avec la base de donnée et le formatage des données (formatage des champs de date, structuration des données pour correspondre à la structure de la base de données, ...).

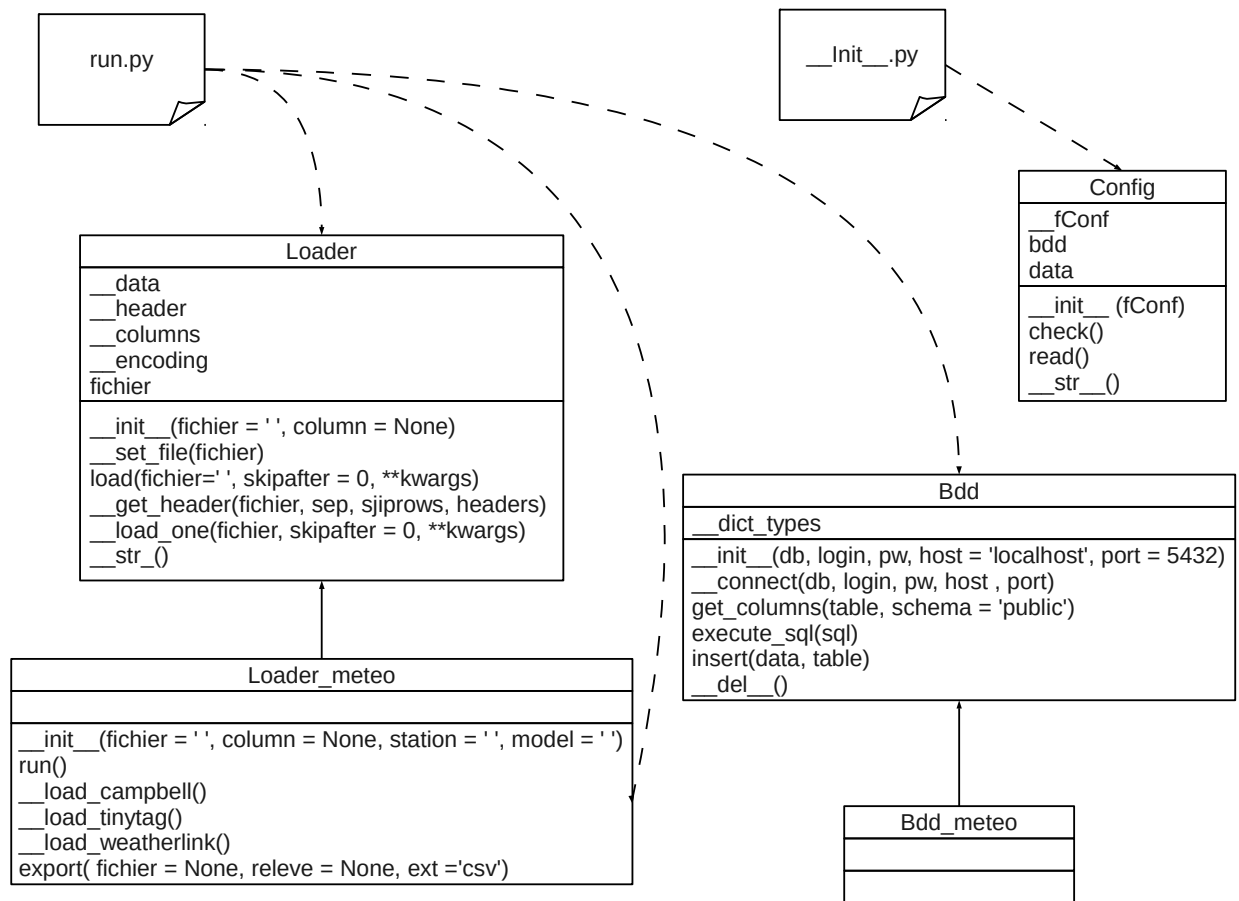


Figure 8: modèle objet du script d'import

À l'heure actuelle, il est nécessaire d'indiquer au script des informations concernant la station d'origine des données, comme l'identifiant de la station ou le formatage du fichier de données. Cependant ces données pourrait être récupérées à partir de l'entête du fichier source. Alban à commencer à effectuer des modifications dans ce sens sur le code et il est possible que je l'y rejoigne d'ici la fin du stage.

III 5 L'interface de consultation

Pour l'instant l'interface de consultation est encore en cours de développement, sa finition serra l'objectif principal durant les deux semaines de stage supplémentaires dont je dispose.

L'aspect graphique de l'interface, réalisée en HTML 5 et CSS 3, ainsi que les réactions aux actions de l'utilisateur, utilisant JavaScript, sont achevés. Il me reste toutefois à produire la partie du code en PHP relative à l'interaction avec la base de données. L'interface ne peut donc pas encore afficher de valeurs de la base.

Lorsque l'interface de consultation serra achevée, il me faudra également finir l'interface administrateur qui est à peu près au même niveau de complétion.

Il est possible que, suite à l'ajout d'un volume relativement important de données dans l'interface, la disposition graphique de celle-ci soit amenée à changer légèrement .

Nombre de stations :

Base de données météorologique

Données souhaitées :

Température Pression Précipitations
 Humidité Vitesse du vent Direction du vent

Agréation et période :

Date debut : H00

Date fin : H00

Plage Horaire : H00
 H00

Horaires Quotidiennes Mensuelles Normales

Insertion des données ici

csv excel

Observatoire de Rennes, Zone Atelier Arctique, CIIRS, LETG, UNIVERSITÉ RENNES

Figure 9: Interface (temporaire) de consultation

III 6 Documentation

Une fois l'interface de consultation achevée, je devrais réaliser un manuel d'administrateur afin de permettre aux opérateur du réseau de se servir de la base de données aisément. Un manuel utilisateur n'est pas requis car l'interface se doit d'être intuitive, elle est donc en hors de ses objectifs si un utilisateur ne s'y retrouve pas et demande de l'aide.

Enfin je dois rédiger un document informatif sur la structure de mon travail afin qu'il soit repris par d'autres après moi car le projet est loin d'être achevé.

IV Bilan

IV 1 Objectifs remplis

La base de données est fonctionnelle. Le script de création est disponible et peut être déployé sur toute machine ayant PostgreSQL et PostGis. Le script d'import fonctionne.

IV 2 Possibilités d'évolution

Le stage n'étant pas fini, on peut considérer l'achèvement des interfaces de consultation et d'administration comme une possibilité d'évolution à très court terme.

Pour la suite il est possible d'ajouter de nouveaux formats de données à la base et de l'utiliser afin d'alimenter d'autres bases ou application (notamment Indigeo).

On peut également envisager que les systèmes d'automatisation de relève de données, lorsqu'ils seront achevés, communiquent directement avec la base de données au lieu de seulement envoyer les fichiers sources à un administrateur de la base.

IV 3 Formation technique acquise

Durant ce stage j'ai appris à utiliser un nouveau SGBD, PostgreSQL, qui permet des modélisations différentes des SGBD rencontrés lors de ma formation.

J'ai également assimilé les bases du langage python et de deux bibliothèques python populaires (Pandas et PqycoPg2).

J'ai pu manipuler et m'améliorer avec les langages du web (HTML, CSS, JavaScript), vis-à-vis de qui je n'étais pas très à l'aise au début de ce stage.

Toutes ces technologies me serviront probablement par la suite.

IV 4 Bilan personnel

Ce stage fut pour moi l'occasion d'expérimenter de nouvelles manières de travailler en équipe, sur des parties spécifiques, tout en jouissant de périodes de totale autonomie.

J'ai également pu avoir un aperçu du travail dans un milieu professionnel hiérarchiquement complexe et du travail dans le milieu de la recherche.

Ce stage m'a également permis d'appréhender les difficultés inhérentes au travail avec des capteurs physiques.

Tous ces éléments me permettent de cerner un peu mieux le monde du travail et me serviront probablement à m'orienter au mieux quand je devrais y entrer.

Bibliographie

Papier :

Hugues Bersini, *La programmation orientée objet Cours et exercices en UML2 avec Java, C# 4, C++, Python, PHP5 et LinQ 5E édition*, éditions EYEROLLES (ISBN 978-2-212-12806-2)

Wes McKinney, *Python for data analysis*, éditions O'REILLY (ISBN 978-1-449-31979-3)

Linux Magazine n°77 : la programmation orientée objet en 6 jours

Web :

<http://www.postgresql.org>

<https://www.python.org/>

<http://pandas.pydata.org/>

<https://pypi.python.org/pypi/psycpg2>

<http://stackoverflow.com>

<http://www.w3schools.com>

<https://letg.univ-nantes.fr>

<https://osur.univ-rennes1.fr>

Stage de conception d'une base de donnée météorologique.

LETG Rennes Costel est une branche de l'UMR LETG localisée, à Rennes, qui mène des recherches dans le domaine de l'environnement, la climatologie, la télédétection et l'écologie sur la ZA Armorique, dans le cadre de l'OSUR. Le Costel gère un réseau de stations météorologiques dispersé sur la ZAA. Mais les données recueillies de cette manière ne sont pas centralisées et il est donc actuellement difficile de rechercher une information dans cet énorme quantité de fichiers textes.

Pour ce stage je dois concevoir une base de données météorologique chargée d'accueillir les données collectées par le Costel depuis 1996 et les afficher dans une interface web. En plus de ceci, je dois également coder des scripts afin de remplir cette base de données avec les données provenant des fichier texte actuellement en usage.

Mots clés :

SQL, PostgreSQL, Python, Base de données, Web, HTML, CSS, JavaScript, PHP, Environnement, Climatologie, Météorologie, OUSR, ZA Armorique,

Internship in meteorological database design

LETG Rennes Costel is a branch of LETG, localized in Rennes, which lead research about environment, climatology, remote sensing and ecology on the ZA Armorique as part of the OSUR. Costel manage a network of weather stations all around the ZAA, but the data gather this way aren't centralized and it's actually quite hard to get an information from this huge amount of text files.

For my stage I have to design a meteorological database for store all the data collect by Costel from 1996 and display it in a web interface. In addition, I also have to program some scripts to fill the database with the data coming from the text files actually in use.