

## Cycle Certifiant Data scientist

Durée: 8 jours

3870 €

### Public:

Chefs de projet, data scientists, statisticiens, développeurs souhaitant comprendre les impacts du BigData au niveau du traitement des données, des architectures, de l'organisation de la DSI, et maîtriser les étapes de traitement des données, depuis l'analyse de la qualité des données, les processus de Machine Learning, jusqu'à la mise en oeuvre des différents outils.

### Objectifs:

Comprendre les concepts et les apports des technologies BigData, savoir définir les étapes de préparation des données, et les algorithmes de Machine Learning. Connaître les apports des outils comme Hadoop, Spark, le rôle des différents composants, la mise en oeuvre pour du stream processing, des traitements de Machine Learning, des calculs statistiques avec les graphes.

### Connaissances préalables nécessaires:

Connaissance des bases des systèmes d'information, et notions de calculs statistiques.

### Programme:

## Cycle Certifiant Data scientist

- Introduction** : A l'origine du BigData : traitement de volumes importants de données non structurées, traitements optimisés de flux de données au fil de l'eau, liés aux nouvelles technologies et aux nouveaux usages. Domaines concernés : recherche scientifique, médical, e-commerce, sécurité, prédictif, ... Apports des évolutions techniques sur différents aspects : stockage, indexation/recherche, calcul. Concepts clés : ETL, Extract Transform Load, CAP, 3V, 4V, données non structurées, prédictif, Machine Learning. Quelques applications : Watson (IBM), Amazon Rekognition Le positionnement des technologies de cloud, BigData et noSQL, de data-mining. Eléments d'architecture. Gouvernance des données : importance de la qualité des données, fiabilité, durée de validité, sécurité des données Aspects législatifs : sur le stockage, la conservation de données, etc ... sur les traitements, la commercialisation des données, des résultats
- Stockage distribué** : Caractéristiques NoSQL Les différents modes et formats de stockage. Besoin de distribution. Définition de la notion d'élasticité. Principe du stockage réparti : Définitions : réplication, sharding, gossip protocol, hachage, Systèmes de fichiers distribués : GFS, HDFS, Ceph Les bases de données : Cassandra, HBase, MongoDB, CouchBase, Riak, BigTable, ..
- Calcul et restitution** : Apport des outils de calculs statistiques Langages adaptés aux statistiques, liens avec les outils BigData. Outils de calcul et visualisation : R, SAS, Spark, Tableau, QlikView, ... Caractéristiques et points forts des différentes solutions.

## Cycle Certifiant Data scientist

- Evolutions** : Liens vers les nouveaux métiers : Hadoop scientists, Data scientists, CDO.  
Analyse des données au service de l'entreprise  
Rôle de la DSI dans la démarche BigData.  
Ouverture sur l'OpenData : principe, la démarche publique, les licences.  
Exemple : portail data.gouv.fr  
Les offres Saas BigData comme Google BigQuery.  
Les limites. Les nouveautés annoncées.
- Présentation hadoop** : Historique du projet hadoop  
Les fonctionnalités : stockage, outils 'extraction, de conversion, ETL, analyse, ...  
Exemples de cas d'utilisation sur des grands projets.  
Les principaux composants :  
HDFS pour le stockage et YARN pour les calculs.  
Les distributions et leurs caractéristiques (HortonWorks, Cloudera, MapR, GreenPlum, Apache, ...)
- L'architecture** : Terminologie : NameNode, DataNode, ResourceManager  
Rôle et interactions des différents composants  
Présentation des outils d'infrastructure : ambari, avro, zookeeper;  
de gestion des données : pig, oozie, tez, falcon, pentaho, sqoop, flume;  
d'interfaçage avec les applications GIS;  
de restitution et requêtage : webhdfs, hive, hawq, impala, drill, stinger, tajo, mahout, lucene, elasticSearch, Kibana  
Les architectures connexes : spark, cassandra
- Exemples interactifs** : Démonstrations sur une architecture Hadoop multi-noeuds.  
Mise à disposition d'un environnement pour des exemples de calcul  
Travaux pratiques :  
Recherches dans des données complexes non structurées.
- Applications** : Cas d'usages de hadoop.  
Calculs distribués sur des clusters hadoop

## Cycle Certifiant Data scientist

- Zoom sur les données** : Format, volumes, structures, ... et les requêtes, attentes des utilisateurs.  
Etapes de la préparation des données.  
Définitions, présentation du data munging  
Le rôle du data scientist.
- Gouvernance des données**: Qualité des données.  
Transformation de l'information en donnée. Qualification et enrichissement.  
Sécurisation et étanchéité des lacs de données.  
Flux de données et organisation dans l'entreprise.  
De la donnée maître à la donnée de travail. MDM.  
Mise en oeuvre pratique des différentes phases : nettoyage, enrichissement, organisation des données.
- Traitements statistiques de base** : Introduction aux calculs statistiques. Paramétrisation des fonctions.  
Applications aux fermes de calculs distribués. Problématiques induites. Approximations. Précision des estimations.
- Data Mining** : Besoin, apports et enjeux.  
Extraction et organisation des classes de données.  
Analyse factorielle.
- Machine Learning**: Apprentissage automatique  
Définition, les attentes par rapport au Machine Learning  
Les valeurs d'observation, et les variables cibles.  
Ingénierie des variables.  
Les méthodes : apprentissage supervisé et non supervisé  
Classification des données,  
Algorithmes : régression linéaire, k-moyennes, k-voisins, classification naïve bayésienne, arbres de décision, forêts aléatoires, etc ..  
Création de jeux d'essai, entraînement et construction de modèles.  
Prévisions à partir de données réelles. Mesure de l'efficacité des algorithmes. Courbes ROC.  
Parallélisation des algorithmes. Choix automatique.

## Cycle Certifiant Data scientist

IA	: Introduction aux réseaux de neurones. Réseaux de neurones à convolution. Modèles de CNN. Les types de couches : convolution, pooling et pertes. L'approche du Deep Learning. Deeplearning4j sur Spark.
Les risques et écueils	: Importance de la préparation des données. L'écueil du "surapprentissage".
Visualisation des données	: L'intérêt de la visualisation. Outils disponibles, Exemples de visualisation avec R et Python
Traitement de données avec Spark	: Présentation Spark, origine du projet, apports, principe de fonctionnement Langages supportés.
Premiers pas	: Utilisation du shell Spark avec Scala ou Python Gestion du cache
Règles de développement	: Mise en pratique en Java et Python Notion de contexte Spark Différentes méthodes de création des RDD: depuis un fichier texte, un stockage externe. Manipulations sur les RDD (Resilient Distributed Dataset) Fonctions, gestion de la persistance.
Cluster	: Différents cluster managers : Spark en autonome, avec Mesos, avec Yarn, avec Amazon EC2 Architecture : SparkContext, Cluster Manager, Executor sur chaque noeud. Définitions : Driver program, Cluster manager, deploy mode, Executor, Task, Job Mise en oeuvre avec Spark et Amazon EC2 Soumission de jobs, supervision depuis l'interface web
Intégration hadoop	: Travaux pratiques avec YARN Création et exploitation d'un cluster Spark/YARN.

## Cycle Certifiant Data scientist

- Support Cassandra** : Description rapide de l'architecture Cassandra. Mise en oeuvre depuis Spark.  
Exécution de travaux Spark s'appuyant sur une grappe Cassandra.
- Spark et SQL** : Objectifs : traitement de données structurées, .  
Optimisation des requêtes.  
Mise en oeuvre des Dataframes et DataSet.  
Comptabilité Hive  
Travaux pratiques:  
extraction, modification de données dans une base distribuée  
L'API Dataset :  
disponible avec Scala ou Java.  
Collections de données distribuées.  
Exemples.
- Streaming** : Objectifs , principe de fonctionnement : stream processing.  
Source de données : HDFS, Flume, Kafka, ...  
Notion de StreamingContexte, DStreams, démonstrations  
Travaux pratiques : traitement de flux DStreams en Java.
- Machine Learning**: Fonctionnalités : Machine Learning avec Spark,  
algorithmes standards,  
gestion de la persistance,  
statistiques.  
Support de RDD.  
Mise en oeuvre avec les DataFrames.
- Spark GraphX** : Fourniture d'algorithmes, d'opérateurs simples pour des calculs statistiques sur les graphes  
Travaux pratiques :  
exemples d'opérations sur les graphes.