



**HAL**  
open science

# Cloud computing appliqué au traitement multimodal d'images in situ pour l'analyse des dynamiques environnementales

Elisabeth Ranisavljević

► **To cite this version:**

Elisabeth Ranisavljević. Cloud computing appliqué au traitement multimodal d'images in situ pour l'analyse des dynamiques environnementales. Géographie. Université Toulouse le Mirail - Toulouse II, 2016. Français. NNT : 2016TOU20128 . tel-01947202

**HAL Id: tel-01947202**

**<https://theses.hal.science/tel-01947202>**

Submitted on 6 Dec 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Université  
de Toulouse

# THÈSE

En vue de l'obtention du

## DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse - Jean Jaurès*

Cotutelle internationale *École Internationale des Sciences du Traitement de l'Information*

---

---

Présentée et soutenue le *12/12/2016* par :

**ELISABETH RANISAVLJEVIĆ**

*Cloud computing* appliqué au traitement multimodal d'images *in situ*  
pour l'analyse des dynamiques environnementales

---

---

### JURY

THANH THUY NGUYEN  
JEAN-PIERRE BARBOT  
DENIS MERCIER  
NISRINE FORTIN  
CAMDAVANT  
MYRTILLE MOREAU

FLORENT DEVIN  
DOMINIQUE LAFFLY  
YANNICK LE NIR

Professeur des Universités  
Professeur des Universités  
Professeur des Universités  
Enseignante-Chercheuse

Docteur en géographie  
Ingénieur géoinformation  
Enseignant-Chercheur  
Professeur des Universités  
Enseignant-Chercheur

Président du Jury  
Rapporteur  
Rapporteur  
Examinatrice  
Examinatrice  
Co-Encadrant  
Directeur de Thèse  
Co-directeur de Thèse

---

### École doctorale et spécialité :

*TESC : Géographie et aménagement*

### Unité de Recherche :

*Géographie de l'Environnement (UMR 5602)*

### Directeur(s) de Thèse :

*Dominique LAFFLY et Yannick LE NIR*

### Rapporteurs :

*Jean-Pierre BARBOT et Denis MERCIER*

Learn from yesterday, live for today, hope  
for tomorrow. The important thing is not  
to stop questioning.

---

*Albert Einstein*

# Remerciements

Try not to become a man of success, but rather try to become a man of value.

---

*Albert Einstein*

It is the supreme art of the teacher to awaken joy in creative expression and knowledge.

---

*Albert Einstein*

Je souhaite remercier tous ceux qui m'ont accompagné durant ces années de doctorat.

Un grand merci à Dominique LAFFLY, Professeur de Géographie à l'université de Toulouse Jean-Jaurès, pour avoir accepté de diriger cette thèse et d'y avoir vu tout son potentiel futur. Merci de m'avoir fait découvrir le monde de la géographie, la problématique des milieux polaires, mais également la problématique de gestion et traitement des données géographiques.

Un grand merci également à Yannick LE NIR, co-directeur de cette thèse, ainsi que Florent DEVIN, examinateur de cette thèse, tout deux enseignant-chercheur en Informatique à l'EISTI. Depuis le début de mes études, j'ai eu la chance de les avoir comme professeur. Non seulement ils ont su me donner le goût de l'informatique mais ils ont également su susciter en moi l'envie d'aller plus loin, de me surpasser, et je les remercie de m'avoir encouragé à poursuivre dans la recherche. Sans eux, ce travail n'aurait jamais pu être possible. Je tiens à vous remercier tout les deux pour votre disponibilité, votre présence et votre réactivité tout au long de cette thèse.

Je remercie également Didier GALOP, directeur de recherche au CNRS et directeur du Laboratoire GEODE, pour m'avoir accueilli au sein du laboratoire.

Je remercie les membres du jury d'avoir accepté d'évaluer cette thèse, et tout particulièrement les rapporteurs, Jean-Pierre BARBOT et Denis MERCIER, pour leur lecture très attentive et de leurs remarques précieuses. Je suis également reconnaissante aux membres de mon comité de thèse, Patrick REUTER et Jean-Michel FRIEDT, pour le temps qu'ils m'ont accordé, pour leurs remarques pertinentes et l'intérêt qu'ils ont porté à ce travail.

Mes remerciement s'adressent également à Ali MOHAMMAD-DJAFARI, du Laboratoire des Signaux et Systèmes de Centrale-Supélec, pour m'avoir accueilli pendant un mois dans les locaux à Gif-sur-Yvette et pour avoir partagé son savoir et ses idées avec moi.

Un grand merci à Rémi VERNAY, Alexandre ANGEL et Christophe DAUPHIN pour toute leur aide, leur temps, leur soutien, leur réactivité et leur conseil dans la mise en place des architectures de tests, des outils informatiques utilisés et des moyens mis en place. Merci également à Nisrine FORTIN CAMDAVANT et à Astrid JOURDAN pour leur aide dans les spécificités mathématiques et pour leur soutien tout le long de ce travail. Un grand merci à Magdalène COYNAULT pour les différentes relectures et corrections de mon anglais pour les différentes communications écrites. Merci à Myrtille MOREAU et à Nisrine FORTIN CAMDAVANT pour leur soutien et leur aide dans les derniers moments difficiles de rédaction. Merci à Peio LOUBIERE, Sonia YASSA, Zoubeida ABDELMOULA, et aux autres membres du laboratoire, mais également à tous mes collègues de l'EISTI, Nadège, Krystel, Christianne, Laurence, Sébastien, Romain, Pierre, ... pour leur bonne humeur et leur motivation journalière.

Merci également à mes étudiants, pour leur patience avec moi les jours où la fatigue était très présente, mais également pour leur motivation dans les différents projets informatique qui étaient inspirés des travaux de ma thèse. Merci pour l'intérêt porté, pour la motivation, pour les idées plus ou moins constructives et faisables, et merci tout particulièrement à Dupond et Dupont *and co.* pour les moments humoristiques partagés.

À mes amis, JJ, Meari, Arii, Sunny, Joy, Derrick, Sarah, Manue, Nico, Ludymiel, aux familles AVELIN, DUTREY, PEREZ, et à tous les autres (vous citer tous serait trop long) qui m'ont permis d'avoir une vie à côté de ma thèse, qui m'ont permis de respirer et rigoler malgré les passes difficiles.

Merci à mes parents, à mon frère et ma soeur, à mes cousins pour leur intérêt, leur soutien mais aussi pour les moments de détente et de folie. Merci à mon neveu Eliott et ma nièce Charlotte pour leur innocence et leur joie de nourrisson.

Et enfin, j'adresse une pensée particulière et profonde à celui sans qui tout cela aurait été impossible.  
Un grand merci.

# Sommaire

Remerciements	iii
Introduction générale	1
I GÉOGRAPHIE ET <i>Cloud computing</i> , UN NOUVEL HORIZON	5
1 Approche géographique de l'évolution des paysages polaires dans le cadre du changement climatique contemporain	9
2 Le cloud aux limites techniques de la géographie	19
3 Le projet Hydro-Sensor-FLOWS, un glacier sous surveillance	27
4 La variété, l'hétérogénéité et la quantité des données : une base de données <i>cloud</i> adaptée	43
5 Analyse de l'évolution de la couverture neigeuse par photo <i>in situ</i>	51
6 La composition de services de traitements géographiques	65
7 La géographie, laboratoire d'essai à grandeur nature pour le <i>cloud</i>	77
II L'ARCHITECTURE ET LE MODÈLE	79
8 Modèle d'application dans le <i>cloud</i>	83
9 L'orchestration des services	93
III RÉSULTATS	105
10 Comparaison des architectures	109
11 Réduction des dimensions des données	117

<b>Conclusion générale</b>	<b>142</b>
ANNEXE	145
<b>Bibliographie</b>	<b>149</b>
<b>Liste des acronymes</b>	<b>163</b>
<b>Glossaire</b>	<b>165</b>
<b>Table des figures</b>	<b>168</b>
<b>Table des matières</b>	<b>170</b>

# Introduction générale

The true sign of intelligence is not  
knowledge but imagination.

---

*Albert Einstein*

## Contexte général

L'analyse des paysages, des dynamiques et des processus environnementaux nécessite de relever régulièrement des données sur les sites étudiés. Les types de données sont assez variés, telles que climatiques, pluviométriques, hydrologiques, etc. Certaines données demandent une intervention sur place telles que la composition des plantes, le prélèvement des sols ou encore la densité de la neige. Enfin, les images de télédétection (ou de pseudo-satellites comme les *Unmanned Aerial Vehicles* (AUV)) fournissent, à intervalles de temps plus ou moins courts, une vue ubiquiste des paysages. Ces images offrent une résolution spatiale très fine (d'une dizaine de centimètres à quelques mètres) et une variété spectrale couvrant les domaines visibles et infrarouge du spectre électromagnétique.

Pour étudier chacune de ces données, des équipes pluridisciplinaires se forment afin de pouvoir répondre aux besoins d'extraction et d'analyse de données : géologiques, chimiques, physiques, biologiques, etc. Au fil du temps, les données acquises et extraites augmentent en nombre, et les nouvelles technologies permettant la mise en place de nouveaux matériels pour faire une acquisition plus précise, plus régulière et à moindre coût, génèrent de plus en plus de données, autant en nombre qu'en volume. Face à une telle masse d'information, seule l'informatique est capable de stocker et de traiter l'ensemble de ces informations.

## Motivations et objectifs

Le traitement de l'information varie en fonction de la donnée traitée ainsi que de l'information à extraire. Dans le cadre de l'analyse des dynamiques environnementales, il y a des centaines de manières de traiter les informations. Dans le cadre de cette thèse, nous allons nous focaliser sur l'analyse des dynamiques d'un glacier situé en Arctique, et tout particulièrement sur l'analyse de l'évolution de la couverture neigeuse de ce glacier, menée dans le cadre du programme Hydro-Sensor-FLOWS (HSF) de 2006 à 2010.



Le suivi du manteau neigeux nécessite une acquisition régulière de la vue du glacier. Les données télémétriques n'étant pas suffisantes en nombre pour des questions de coût et de fréquence, elles ont été complétées par des photos *in situ*. Contrairement aux données télémétriques, les photos *in situ* sont acquises en vue cavalière et ne correspondent qu'à une vue partielle du glacier. Elles nécessitent un traitement complémentaire afin d'obtenir un équivalent de la donnée télémétrique.

Il existe déjà de nombreux Système d'Information Géographique (SIG) qui permettent de stocker, traiter et analyser des données géographiques (ArcGIS, ERDAS, QGIS, etc). Cependant, ils ne permettent pas de traiter et analyser des informations provenant de photos *in situ*, et encore moins quand la quantité de photos est importante. La première problématique de cette thèse est : **Comment automatiser les traitements tout en restant assez générique afin de pouvoir traiter n'importe quelle image *in situ* (provenance et perturbations) ?**

En effet, les données avec lesquelles nous allons pouvoir effectuer nos tests proviennent du programme HSF, et sont donc des données de glaciers. Les photos *in situ* proviennent d'appareils photo qui ont été placés sur le glacier et donc sont perturbées par l'environnement extrême de l'Arctique. Le principe d'acquisition de photos *in situ* peut se mettre en place dans d'autres domaines que celui de l'analyse de dynamiques de glaciers. Et même si les photos *in situ* ne sont pas perturbées par les conditions extrêmes de l'Arctique, d'autres facteurs peuvent exister et être perturbateurs.

L'informatique a également évolué au cours du temps, avec la création d'ordinateurs personnels, de serveurs, de grilles de calculs et dernièrement de *cloud computing*. Cette évolution a été possible grâce à l'évolution de l'électronique et la mise en place de matériel informatique de plus en plus puissant. En parallèle de cette évolution électronique, la théorie associée à l'informatique a également pu progresser grâce à la mise en place d'algorithmes, qui étaient jusqu'à présent limités par la capacité de la machine, et grâce à la génération de nouvelles problématiques liées à l'évolution technologique (par exemple, la masse de données générée). Afin de traiter les différentes données, les SIG, qui sont puissants pour un traitement manuel des données, atteignent leur limite face au volume généré. La deuxième problématique de cette thèse est : **En quoi le *cloud computing* permet d'automatiser les traitements ? En quoi est-il plus intéressant qu'une simple machine ou qu'un serveur de calcul ?**

L'évolution de la quantité et du volume de données acquises par l'analyse environnementale se rapproche de plus en plus de la problématique du *Big Data*. Même si ces quantité et volume n'atteignent pas encore la dizaine de téraoctets<sup>1</sup> par jour comme peut le faire Facebook, le projet HSF seul a généré environ 500 gigaoctets<sup>2</sup> de données. En plus de la problématique de traitement, nous faisons également face à une problématique de stockage de l'information. La troisième problématique de cette thèse est : **Face à la quantité de donnée, peut-on réduire les données à stocker sans pour autant perdre de l'information ?**

## La thèse dans ses grandes lignes

La structure de cette thèse sur articles va se décomposer en trois parties :

- 
1. 1 téraoctet =  $10^{12}$  octets
  2. 1 gigaoctet =  $10^9$  octets

- la présentation du contexte et des objectifs de recherche face aux deux domaines, la géographie et le *cloud computing* ;
- la conception du système d’application afin de définir notre approche et notre solution technologique ;
- les expériences menées et les résultats obtenus.

Voici le détail de ces différentes parties.

## 1. Géographie et *Cloud computing*, un nouvel horizon

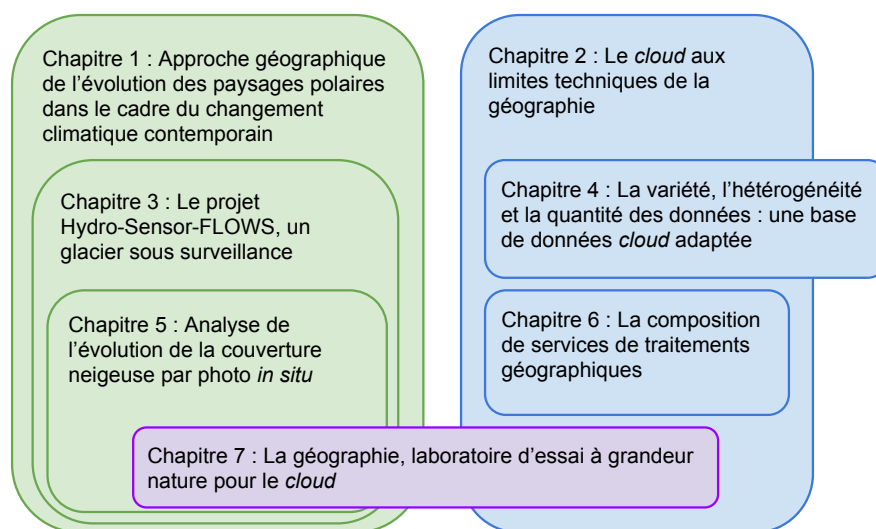


FIGURE 1 – Organisation de la première partie de la thèse. À gauche, en vert, le contexte géographique, où l’on se concentre petit à petit à l’analyse de la couverture neigeuse. À droite, en bleu, le contexte informatique, où l’on détaille différentes notions relatives au *cloud*. En bas, en violet, un lien entre ces deux contextes.

- Chapitre 1 : Nous présenterons le contexte du changement climatique contemporain, et tout particulièrement l’impact de celui-ci sur l’Arctique, notamment ses glaciers.
- Chapitre 3 : Nous rentrerons plus en détails sur un programme scientifique : le projet Hydro-Sensor-FLOWS (HSF). Nous aborderons ses problématiques de recherche et ses différentes analyses associées.
- Chapitre 5 : Parmi les différentes problématiques du programme HSF, nous nous attarderons sur une seule : l’évolution de la couverture neigeuse du glacier. L’analyse de cette évolution a été faite à partir des photos *in situ* qui nécessitent un certain nombre de traitements. Nous détaillerons ses différents traitements nécessaires.
- Chapitre 2 : Au niveau de l’informatique, nous commencerons par présenter le *cloud computing* et nous dresserons l’état de l’art.

- Chapitre 4 : Face à la problématique des données, nous présenterons le principe des bases de données et les équivalents qu'il existe dans le *cloud computing*.
- Chapitre 6 : Afin de traiter l'ensemble des données, nous présenterons le principe des services sur le *cloud computing*, notamment les services web, ainsi que la problématique de composition de services afin d'obtenir le traitement complet de toutes les données.
- Chapitre 7 : Comme nous avons pu le voir, l'interdisciplinarité permet de faire avancer la science : d'un côté en utilisant des nouveaux concepts, et de l'autre en approuvant les différents concepts par des tests réels.

## 2. L'architecture et le modèle

- Chapitre 8 : À travers l'article *A dynamic and generic cloud computing model for glaciological image processing* [Ranisavljević et al., 2014a], nous présenterons le modèle d'architecture mis en place pour répondre à la première problématique.
- Chapitre 9 : À travers l'article *Architecture comparison of an in-situ data processing application applied to glacier mass balance analyse* [Ranisavljević et al., 2013], nous détaillerons notre modèle de composition de services.

## 3. Résultats

- Chapitre 10 : Suite à différents tests effectués sur plusieurs lots de photos traités à travers le prototype de notre modèle d'architecture, nous présenterons les résultats, exposés lors de la conférence *13th International Circumpolar Remote Sensing Symposium* [Ranisavljević et al., 2014b], qui montrent les avantages du *cloud computing*, répondant ainsi à la deuxième problématique.
- Chapitre 11 : Après un court séjour au laboratoire des signaux et systèmes (L2S) à Supélec, nous avons pu mettre en évidence l'existence de méthodes permettant de réduire les dimensions des données, et par cela répondre à la troisième problématique.

Première partie

Géographie et *Cloud computing*,  
un nouvel horizon



Look deep into nature, and then you will  
understand everything better.

---

*Albert Einstein*



# Chapitre 1

## Approche géographique de l'évolution des paysages polaires dans le cadre du changement climatique contemporain

Our task must be to free ourselves by widening our circle of compassion to embrace all living creatures and the whole of nature and its beauty.

---

*Albert Einstein*

It is not that I'm so smart, it's just that I stay with problems longer.

---

*Albert Einstein*

La géographie, littéralement la description de la terre, est axée sur la connaissance du globe depuis l'antiquité par 4 types d'approches [Pattison, 1964, Meentemeyer, 1989, Bailly et al., 2016] :

- les grandes explorations pour **découvrir** les continents ;
- la géométrie pour **mesurer** l'espace terrestre ;
- la cosmographie afin de **situer** la Terre dans les systèmes astronomiques ;
- la cartographie de manière à **représenter** l'espace terrestre.

Ces différentes approches nous montrent deux orientations méthodiques principales dans cette recherche, principalement descriptives : une géométrique et physique (mesurer, situer), une autre attachée aux caractéristiques des lieux (découvrir, représenter).

Au fil du temps et notamment à la Renaissance, la géographie va se définir autour de 4 approches [Boas, 1887, Bailly et al., 2016] :



- la géographie mathématique, composée notamment de la cosmographie et la géométrie ;
- la climatologie, comprenant l'étude des différents gaz, des vents et des pluies ;
- l'hydrographie, intégrant l'océanographie et la fluviologie ;
- la physiographie, correspondant à l'étude des reliefs.

D'après Bernard Varenius, « la géographie est double. Il y a une géographie générale et une spéciale. La première considère la Terre dans son ensemble, en explique les différentes parties et les phénomènes généraux ; la seconde, se guidant sur les règles générales, passe en revue chaque contrée, etc. » On pourrait dire que l'unité intime de la géographie se situe dans le dualisme entre les lois générales et les descriptions particulières qui en sont l'application [de La Blache, 1896].

La géographie s'attache, en particulier, à comprendre les changements qui affectent le paysage. Un paysage est, par définition, une portion d'espace matériel. L'analyse spatiale y joue donc un rôle privilégié [Bertrand, 1972]. « Au lieu de regarder les éléments naturels comme existant d'une manière indépendante, il faut les considérer pour la valeur qu'un certain groupe social leur attribue, et par rapport aux types d'activité du groupe en question dans un espace donné. » [Dematteis, 1971]

À travers une approche systémique, les paysages ne sont pas vu comme étant une entité indépendante mais comme une entité faisant partie d'un environnement et qui s'adapte à cet environnement en fonction des variations de chacun de ses composants.

*DÉFINITION. — Une approche systémique correspond à affirmer que la méthode scientifique est un système d'apprentissage doté d'un sous-système auto-correctif qui lui permet de vérifier la véracité, la transférabilité et la validité des connaissances qu'il produit. [Lapointe, 1993]*

L'étude des paysages fait référence à des éléments ou à des relations tangibles ou intangibles qui ont leur importance dans l'équilibre du paysage [Zube et al., 1982]. Étudier un paysage nécessite d'identifier l'ensemble des éléments qui le composent et la façon dont ils interagissent les uns avec les autres et ce, à la fois dans l'espace et dans le temps. L'approche des paysages nécessite d'intégrer comme variable environnementale le climat et ses évolutions.

## 1.1 Le changement climatique global contemporain



FIGURE 1.1 – L'avance des glaciers semble correspondre à un minimum prolongé d'activité solaire. Ces deux vues du glacier d'Argentière, dans la vallée de Chamonix, une gravure datant de 1850-1860 (à gauche) et une photographie de 1966 (à droite), illustrent un intense recul séculaire. Extrait de "Histoire du climat depuis l'an mil" - Source : Besnier [Besnier and Ladurie, 1968]

## Modifications des systèmes physiques et biologiques et variations de la température en surface pendant la période 1970-2004

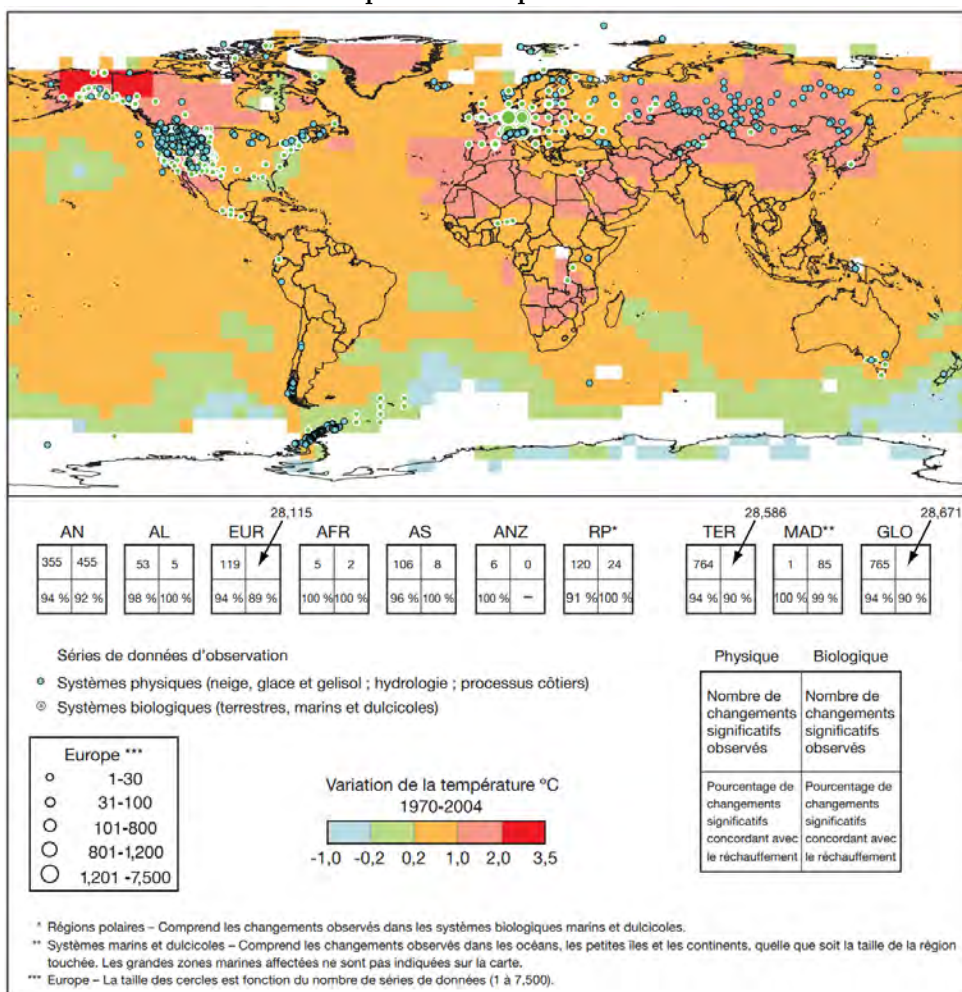


FIGURE 1.2 – Emplacement des changements significatifs relevés dans les séries de données sur les systèmes physiques (neige, glace et sol gelé ; hydrologie ; processus côtiers) et les systèmes biologiques (terrestres, marins et dulcicoles) et variations de la température de l'air en surface pendant la période 1970-2004. Quelques 29 000 séries de données ont été retenues sur les 80 000 publiées dans 577 études, sur la base des critères suivants : 1) se terminer en 1990 ou plus tard ; 2) s'étendre sur une période d'au moins 20 ans ; 3) présenter un changement significatif, dans un sens ou dans l'autre, ayant fait l'objet d'une évaluation dans certaines études. Les séries retenues proviennent de quelques 75 études, dont 70 environ ont été réalisées après la parution du TRE. Sur ces 29 000 séries de données, 28 000 environ sont tirées d'études européennes. Les zones laissées en blanc sont des zones où les données d'observation sont insuffisantes pour qu'il soit possible d'y définir une tendance de la température. Les cases 2x2 indiquent le nombre total de séries de données présentant des changements significatifs (rangée supérieure) et la proportion de celles qui concordent avec le réchauffement (rangée inférieure) pour i) les régions continentales : Amérique du Nord (AN), Amérique latine (AL), Europe (EUR), Afrique (AFR), Asie (AS), Australie et Nouvelle-Zélande (ANZ), régions polaires (PR) ; ii) la planète entière : terres émergées (TER), zones marines et dulcicoles (MAD), globe dans son ensemble (GLO). La somme des différents nombres d'études figurant dans les sept cases des régions continentales (AN, AL, EUR, AFR, AS, ANZ, PR) ne correspond pas au total de la case du globe dans son ensemble (GLO), parce que ces nombres (à l'exception de celui qui concerne les régions polaires) n'incluent pas les études sur les systèmes marins et dulcicoles (MAD). Les grandes zones marines affectées n'apparaissent pas sur la carte. Source : IPCC 2013 [IPCC, 2013]

Le changement climatique global actuel a débuté il y a 150 ans avec la fin du Petit Âge Glaciaire (figure 1.1) [Ladurie, 1959, Mann, 2002, Reynaud and Vincent, 2002]. Les études menées par l'*Intergovernmental Panel on Climate Change* (IPCC) permettent de couvrir l'ensemble des changements liés au climat : la température, les précipitations, l'ensoleillement, l'humidité, la pression atmosphérique et la vitesse du vent [IPCC, 1995].

La température à la surface du globe a augmenté entre 0.2° et 0.8°C depuis la fin du XIX<sup>ème</sup> siècle, et de 0.2° à 0.6°C depuis ces 5 dernières décennies [IPCC, 2014]. Ce réchauffement n'est pas totalement global, car certaines parties du globe se sont refroidies. Les latitudes les plus touchées par ce réchauffement sont celles situées entre 40° et 70°N (figure 1.2). La présence de nuages au niveau de certaines zones terrestres influence de manière générale mais pas forcément globale les températures diurnes et nocturnes : la tendance correspond à une réduction des températures diurnes et une augmentation de celles nocturnes.

Malgré une tendance générale, l'évolution de la répartition des précipitations n'a pas été homogène. En effet, les hautes latitudes de l'hémisphère nord ont subi une augmentation des précipitations, tandis que les régions tropicales et subtropicales ont subi une diminution.

Le comportement de *El Niño*, qui cause des sécheresses ou des inondations dans plusieurs parties du globe, est devenu inhabituel depuis le milieu des années 1970, et tout particulièrement à partir de 1989. Les événements chauds de *El Niño* sont devenus plus fréquents et plus tenaces par rapport à son phénomène opposé *La Niña*. Ceci pourrait expliquer les variations de précipitation dans ces zones subtropicales.

## 1.2 L'Arctique face aux changements climatiques

Le changement climatique impacte particulièrement les zones froides et spécialement l'Arctique, entraînant notamment un accroissement moyen de 0.8°C de la température globale depuis ces cinq dernières décennies, et accélère la fonte des neiges, des glaciers et des banquises, changeant la quantité de neige pendant l'hiver, raccourcissant la durée de l'hiver et remodelant les paysages polaires (figure 1.3)[Barnett et al., 2005].

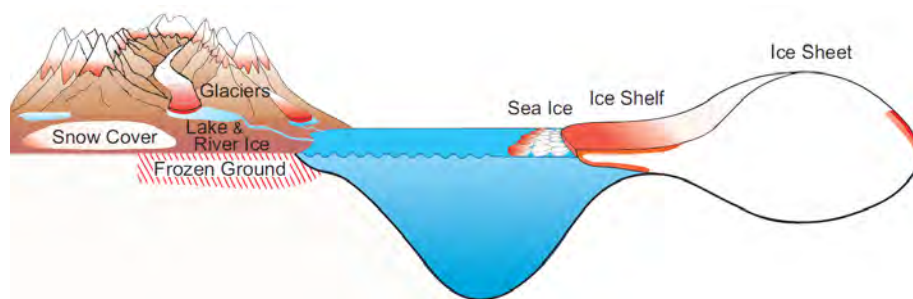


FIGURE 1.3 – Les différents composants de la cryosphère. Les parties particulièrement sensibles au réchauffement climatique sont colorées en rouge. Source : IPCC 2013 [IPCC, 2013]

La région de l'Arctique est définie par sa position dans le cercle Arctique (latitude 66°N), et par les données climatiques et environnementales, comme la courbe isotherme des 10°C au mois de juillet, qui marque la limite au-delà de laquelle les arbres ne poussent plus (figure 1.4). Elle est constituée principalement d'un océan gelé (banquise) entouré de terres très froides (toundra), et est



FIGURE 1.4 – L’isotherme à 10°C du mois de juillet (ligne rouge) qui définit la limite de la région arctique. Source : *World Factbook*

marquée par ses hivers froids (température moyenne de  $-40^{\circ}\text{C}$ ) et ses précipitations solides faibles (hauteur moyenne inférieure à 50 cm).

Le changement climatique qui affecte l’Arctique s’est accéléré vers le milieu des années 1970. Certains changements, comme le gel retardé ou le dégel anticipé des lacs et des rivières [Magnuson et al., 2000], sont le reflet de l’augmentation des températures de l’air [Chapman and Walsh, 1993, Serreze et al., 2000, Serreze and Francis, 2006].

De nombreuses recherches ont mis en évidence les changements climatiques et leurs conséquences sur les milieux arctiques. Des observations de température sur le long terme montrent que la tendance du réchauffement s’est accentuée à travers l’Arctique [Chapin et al., 2005, Hinzman et al., 2005, Overland et al., 2004, Serreze et al., 2000, White et al., 2007]. Les conséquences sur les paysages sont importantes, elle se marquent notamment par la fonte du pergélisol. Le pergélisol de l’Arctique, monitoré par les forages de 10 à 20 m de profondeur, s’est réchauffé de  $2^{\circ}\text{C}$  depuis les 2 à 3 dernières décennies [Romanovsky et al., 2010]. La couverture de glace de l’océan Arctique continue de rétrécir et de s’amincir [Gerland et al., 2008, Holland et al., 2010, Serreze and Francis, 2006, Stroeve et al., 2007]. Ce sont les principaux éléments de la cryosphère qui sont touchés par le

réchauffement global (figure 1.3) [Barry et al., 2011].

Afin de mieux appréhender ces variations et de pouvoir tirer des conclusions sur l'évolution des différents glaciers et zones paraglaciaires (phénomène lié à la fonte de la glace) et périglaciaires (phénomène lié au gel de l'eau) [Mercier, 2000, Oerlemans, 2005], plusieurs études ont été menées sur différents aspects se rapportant aux glaciers.

### 1.3 Les glaciers au cœur du paysage polaire

Un glacier est, d'après Reynaud [Reynaud et al., 1986], un stock d'eau solide permanent à l'échelle humaine, sous forme de neige, de névé ou de glace. Selon Zryd [Zryd, 2001], c'est un système ouvert, pérenne, formé essentiellement de glace naturelle et présentant des signes de mouvement. Il rajoute à la première définition la notion de mouvement. Quant à Francou [Francou and Vincent, 2007], il le définit comme étant une masse de glace d'ampleur hectométrique et plus, permanente à l'échelle humaine, qui se déforme sous son propre poids.

Le glacier, qui est un système ouvert, est continuellement en interaction avec son environnement. Les échanges de matière et d'énergie se font au niveau des surfaces qui l'entourent, glace/air et glace/roche, et les lois physiques qui contrôlent les mécanismes internes du glacier sont à l'origine de son comportement et ses réactions face aux changements ambiants. Le glacier est ainsi continuellement en évolution.



FIGURE 1.5 – Longyearbyen, Spitsberg, 2013. Photo : Ole Joergen

Lorsque, pendant plusieurs années consécutives, la quantité de précipitations solides est supérieure à celle qui fond durant l'été, le glacier peut grandir sous réserve que la neige ne fonde pas entièrement.

Le glacier est en mouvement, il glisse le long de la pente rocheuse sous la force de gravité due à son poids. Il s'étend ainsi sur une surface plus grande que celle où il a pris naissance, se trouvant partiellement dans des conditions de fonte de la glace. Le glacier se compose donc de deux parties : une zone d'accumulation où la neige présente en surface du glacier ne fond pas d'une année sur l'autre, permettant donc à la neige de densifier et de devenir glace au bout d'un certain temps ; et d'une zone d'ablation, où la neige qui tombe chaque hiver ainsi que la glace fondent pendant la saison estivale (figure 1.6).

Quelque soit le glacier, celui-ci est toujours composé de deux zones bien distinctes : la zone d'accumulation et la zone d'ablation, séparées par une ligne d'équilibre glace/névé. Cette distinction repose sur la notion de bilan glaciaire qui prend en considération les entrées (précipitations solides, avalanches) et les sorties (fonte). La ligne d'équilibre est un indicateur qui, par ses changements de position, reflète les variations climatiques. La proportion de la zone d'accumulation d'un glacier par rapport à sa surface totale nous donne également une indication sur sa dynamique. On admet

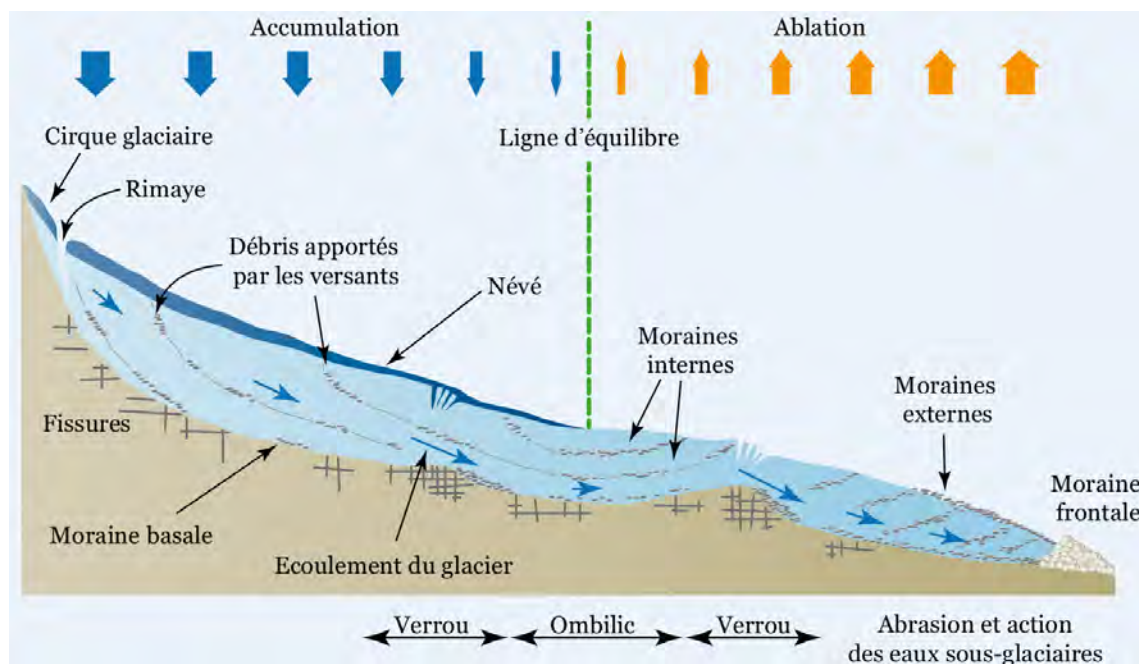


FIGURE 1.6 – Un glacier est composé de deux zones (séparées par une ligne d'équilibre) dont la superficie et l'altitude fluctuent en fonction des conditions climatiques. Ce schéma synthétise son fonctionnement et son action géomorphologique. Source : Bernard 2011 [Bernard, 2011]

que si la zone d'accumulation constitue les deux tiers du glacier, celui-ci est en équilibre [Lliboutry, 1965]. Plus la zone d'accumulation augmente, plus le glacier a tendance à grandir. À l'inverse, plus la zone d'accumulation diminue, plus l'épaisseur et la longueur du glacier ont tendance à rétrécir.

L'indicateur du bilan de masse glaciaire annuel permet de comprendre la relation qui existe entre les fluctuations du climat et des glaciers [Reynaud et al., 1986, Dowdeswell et al., 1997, WGMS, 2008, Bernard et al., 2013a].

## 1.4 Les moraines, paysage du retrait glaciaire

Le retrait des fronts glaciaires laisse à son aval des moraines. La moraine est formée par le matériel qui provient de l'érosion et du dépôt liés au glacier. Le matériel est très hétérogène : il se compose de particules fines comme la farine glaciaire, de roches et jusqu'à de très gros blocs dit erratiques [Agassiz and Bettannier, 1840].

Il existe plusieurs types de moraines qui sont classées suivant leurs positions par rapport au glacier et leurs origines :

**Les moraines latérales** se forment sur les bords du glacier et proviennent des pierres et blocs qui tombent depuis les flancs de la vallée glaciaire qui surplombent le glacier ; quand celui-ci s'est retiré il reste deux cordons de moraines latérales pouvant atteindre la centaine de mètres de haut.

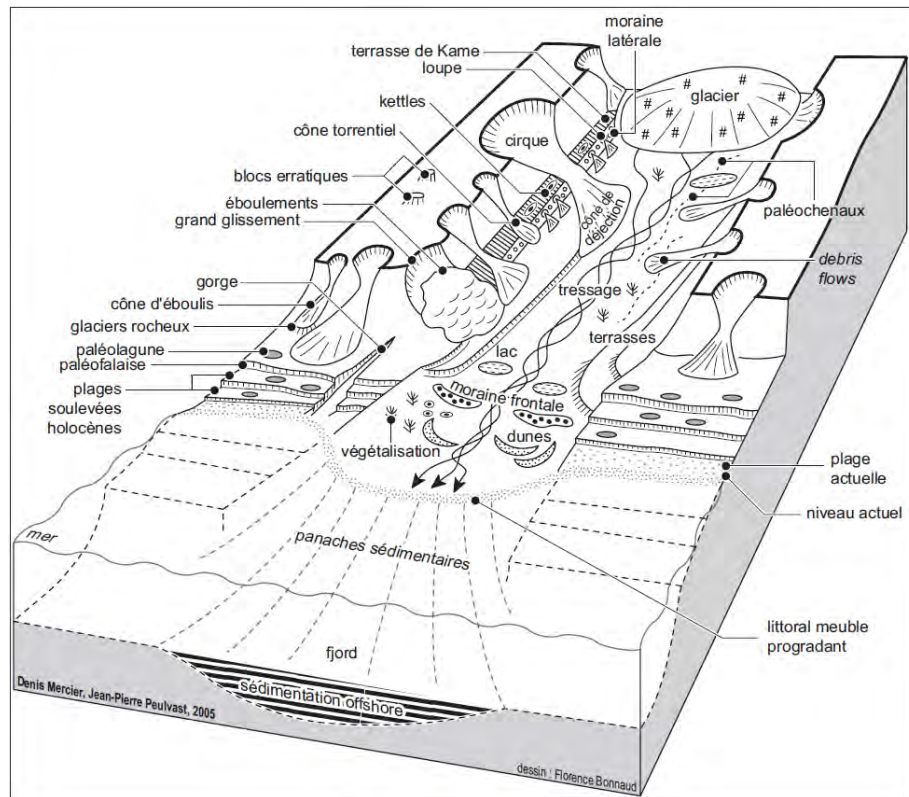


FIGURE 1.7 – Formes et processus paraglaciaires. Source : Mercier 2008 [Mercier, 2007]

**Les moraines médianes** se trouvent vers le milieu du glacier et correspondent à la réunion de deux moraines latérales lorsque deux glaciers se rejoignent.

**Les moraines frontales** sont constituées de blocs tombés sur le glacier qui sont amenés jusqu'au front du glacier et s'y accumulent en permanence. Lorsque le glacier stationne, la moraine frontale s'épaissit ; puis, lorsqu'il recule, la moraine frontale demeure, marquant ainsi l'avancée maximale du glacier.

La présence des moraines permet de témoigner des anciennes positions du glacier. Les moraines peuvent être soumises à différents processus d'érosion notamment ceux liés au gel et dégel, ainsi qu'à l'érosion liée aux écoulements qui peuvent provoquer des coulées de boues, des coulées de débris et des ruissellements qui viennent refaçonner le paysage (figure 1.7). Non seulement le ruissellement favorise l'ablation, mais il favorise également le transport de matières jusqu'à la grève ou jusqu'au fond des fjords [Mercier, 1997, Mercier, 2000, Mercier, 2007].

Les moraines sont de nouveaux espaces propices à la colonisation végétale [Moreau, 2005]. Il est possible de suivre une progression du couvert végétal et de la variété floristique en relation avec les différentes phases de retrait glaciaire et les conditions environnementales locales [Moreau et al., 2005].

## 1.5 Le cas de la recherche au Spitsberg

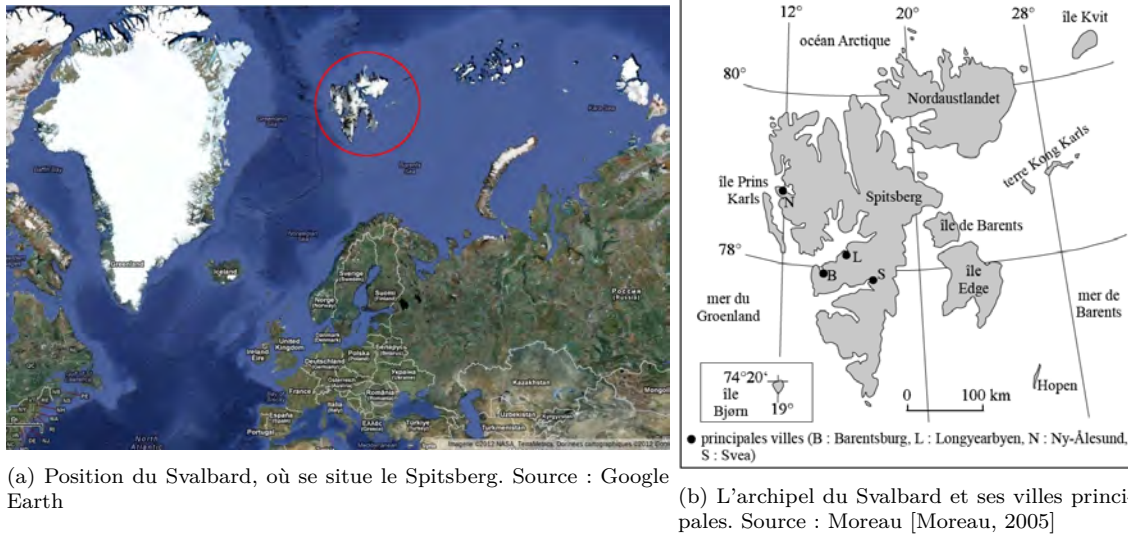


FIGURE 1.8 – Localisation de l'île du Spitsberg.

Le Spitsberg (figure 1.8) est une île connue par les Français depuis bien longtemps. Dès le XVII<sup>ème</sup> siècle, les pêcheurs basques et normands se sont aventurés dans les mers septentrionales afin de ramener le *lard de baleine* [Péguy and Brossard, 1993]. En 1638, sous l'autorité de Richelieu, le basque Vroliq dressa la première carte du Spitsberg, qui fut publiée par le géographe La Varenne lors de son retour d'expédition en 1693, puis reprise par la Cartographie Royale en 1758 (figure 1.9).



FIGURE 1.9 – Les ressources cartographiques exploitables datent du XVII<sup>ème</sup> siècle avec les premières expéditions « scientifiques ». La carte présentée est celle réalisée par la Cartographie Royale, inspirée de l'originale de La Varenne (1758). Ce document donne déjà une idée précise des contours de l'archipel. (Musée National de la Marine). Source : MapAndMaps.com



L'archipel du Svalbard est placé sous souveraineté norvégienne depuis le 9 février 1920 suite au traité concernant le Spitsberg signé à Paris. Ce traité garantit aux différents pays cosignataires des droits d'installation, de prospection et de recherche sur l'archipel du Svalbard. En 1951-1952, Jean Corbel initie les premières missions de terrain en presqu'île de Brøgger. Une dizaine d'années plus tard, la première "base" française – de dimensions 3m×3m – est construite non loin de Ny Ålesund (figure 1.10), permettant d'accueillir, pendant leurs séjours, des équipes de scientifiques.



FIGURE 1.10 – Base française Corbel, proche de Ny-Ålesund, en août 2012.

En 1964, Jean Corbel devient responsable de la structure R.C.P. (Recherches coopératives sur Programme) 42 *Hautes latitudes Arctiques* (CNRS). En continuité de ses recherches sur la morphologie périglaciaire dans l'Arctique [Corbel, 1961] et sur les problèmes de morphologie glaciaire au Spitzberg [Corbel, 1953], Jean Corbel initie des collaborations au sein d'une équipe pluridisciplinaire rassemblant les domaines de l'hydrologie, la chimie, la physique ou la géologie. La R.C.P 42 étudie en 1966 l'influence du froid sur le milieu naturel (morphologie terrestre et littorale, hydrologie, glaciers de montagne, biologie animale, géochimie, géologie polaire, etc.) [Corbel, 1966].

Après la dissolution du programme R.C.P. en 1967 et le décès de Jean Corbel en 1970, les missions au Spitsberg diminuent et la base est abandonnée. En 1974, deux étudiants de Besançon, Thierry Brossard et Daniel Joly, reprennent l'initiative et remettent en fonctionnement cette base en continuant la recherche en géographie au Spitsberg. Les missions se sont succédées jusqu'à nos jours et elles continuent encore, exploitant plus de 50 années d'acquisitions de données quasi-continues. Par exemple, une première étude sur la végétation au niveau des rives plates au Svalbard avait été menée en 1975 par Th. Brossard et a été reprise, 31 ans plus tard, afin d'observer les changements de la végétation sur ce laps de temps [Moreau et al., 2009]. Cet été, une nouvelle mission a permis de continuer cette étude en faisant de nouveaux relevés pour de nouvelles analyses. Un autre suivi concerne le glacier du Lovén Est, situé non loin de la base, qui, depuis les travaux de Madeleine Griselin [Griselin, 1982], a régulièrement fait l'objet d'observations jusqu'à aujourd'hui à travers les programmes HSF et Cryo-Sensors.

Les données acquises depuis le début des travaux de recherche au Spitsberg sont de plus en plus importantes, les moyens d'acquisition sont de plus en plus performants et de plus en plus précis, nécessitant des traitements permettant d'intégrer le nombre croissant d'informations. La géographie se trouve parfois limitée dans ses outils pour l'exploitation d'importantes bases de données.

L'informatique et le recours au *cloud computing* peut apporter des réponses techniques aux limites d'exploitation rencontrées par le géographe.

## Chapitre 2

# Le cloud aux limites techniques de la géographie

Most of the fundamental ideas of science are essentially simple, and may, as a rule, be expressed in a language comprehensible to everyone.

---

*Albert Einstein*

### 2.1 La naissance du *cloud computing*

L'idée principale derrière le *cloud computing* n'est pas nouvelle. Dans les années 1960, Douglas Parkhill décrit dans son livre "*The Challenge of the Computer Utility*" [Parkhill, 1966] un futur où les installations et les traitements informatiques seraient fournis au grand public de la même manière que l'électricité, l'eau ou le gaz. De même, John McCarthy prédit que "*computation may someday be organized as a public utility*". Jusqu'à il y a environ 30 ans, quand les ordinateurs personnels sont arrivés, le fonctionnement classique était d'utiliser les ressources locales pour stocker les données ou faire tourner un programme [Wang et al., 2010]. Depuis ces deux dernières décennies, la génération de données de plus en plus complexes et volumineuses et la nécessité de les traiter ont augmenté. Le calcul haute performance (HPC) nécessaire pour le traitement était limité aux institutions qui avaient les moyens de se payer un superordinateur dédié à ces tâches. Le besoin de faire du HPC à plus petite échelle et à moindre coût a fait émerger vers le milieu des années 1990 le *cluster computing* puis le *grid computing*.

DÉFINITION. — *A cluster is a collection of parallel or distributed computers which are interconnected among themselves using high-speed networks, such as gigabit Ethernet, SCI, Myrinet and Infiniband. They work together in the execution of compute intensive and data intensive tasks that would be not feasible to execute on a single computer. [Sadashiv and Kumar, 2011]*

Les ordinateurs, reliés ensemble, ont permis de générer une puissance de calcul et de stockage

proche du superordinateur mais à moindre coût. L'émergence des *clusters* a été poussée par plusieurs projets académiques comme Beowulf [Becker et al., 1995], Berkeley NOW [Anderson et al., 1995] et HPVMM [Chien et al., 1997]. Le *cluster* est vu par l'utilisateur comme étant une seule machine virtuelle. Les traitements demandés par l'utilisateur sont reçus puis distribués sur chacun des ordinateurs formant le *cluster*.

Un *cluster* étant interne à une institution, il se peut que l'un soit totalement chargé en calcul alors qu'un autre soit totalement libre. Le *grid computing*, qui a également émergé dans le domaine académique, correspond à l'utilisation à distance de ressources non utilisées lorsque les ressources locales le sont totalement.

DÉFINITION. — *Grid technologies support the sharing and coordinated use of diverse resources in dynamic "virtual organizations" – that is, the creation, from geographically and organizationally distributed components, of virtual computing systems that are sufficiently integrated to deliver desired qualities of service. [Foster et al., 2001]*

Plusieurs chercheurs ont produit des exemples de *grid computing*, comme TeraGrid [Reed, 2003, Beckman, 2005], Open Science Grid [Pordes et al., 2007] ou caBIG [Fenstermacher et al., 2006], qui fournissent non seulement une puissance de calcul mais également des données et des applications à la demande.

Avec le succès d'Internet, le *cloud computing* est apparu vers la fin de 2007. Il fournit un ensemble de ressources, tout comme le *grid*, appelées *services*, qui sont accessibles à la demande via Internet.

Le *cloud* est actuellement à la mode et le nombre de services fournis via le *cloud* augmente grandement. De nombreux grands acteurs de l'industrie du logiciel et d'Internet comme Microsoft, Google ou Amazon, fournissent et développent des services [Armbrust et al., 2010, Hilley, 2009, Lin et al., 2009, Murray, 2009, Qian et al., 2009, Weiss, 2007] afin de procurer des plateformes plus puissantes, plus fiables et plus rentables à leurs clients [Zhang et al., 2010]. De même, les entreprises cherchent à remodeler leurs modèles économiques afin de bénéficier de ce nouveau paradigme qui leur offre plusieurs avantages que nous allons détailler dans la prochaine section.

## 2.2 Le *cloud*, qu'est ce que c'est ?

Pendant plusieurs années, le concept du *cloud computing* était encore abstrait et vague, et chacun l'a défini à sa façon. Nous pouvons voir dans [Geelan et al., 2008] que vingt-et-un experts ont déjà donné leur définition. Depuis 2011, le NIST (*National Institute of Standards and Technology*) a formalisé la définition du *cloud computing* ainsi :

DÉFINITION. — *Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. [Mell and Grance, 2011]*

Ian Foster, en comparant le *Grid* et le *Cloud* [Foster et al., 2008], souligne que **oui**, la vision est la même pour les deux concepts : diminuer les coûts de calcul, augmenter la fiabilité et augmenter la flexibilité, **mais non**, ils sont tout de même différents, notamment par l'accès au *cloud* en payant

uniquement ce qui sera utilisé et en utilisant uniquement les ressources nécessaires. [Sadashiv and Kumar, 2011] fait toute une comparaison détaillée entre le *cluster*, le *grid* et le *cloud*. **Cependant, oui** les problématiques scientifiques restent les mêmes : la gestion des équipements, la découverte et l'utilisation des services, la sécurité des données. En figure 2.1, nous pouvons avoir un aperçu des relations et des intersections entre ces différents concepts.

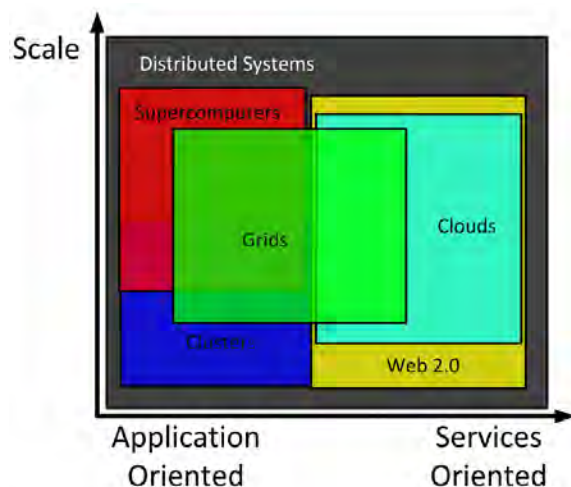


FIGURE 2.1 – Grids et Clouds [Foster et al., 2008]

Afin de compléter la définition du *cloud*, le NIST explicite également différentes caractéristiques essentielles qui le définissent [Mell and Grance, 2011] :

- **On-demand self-service** : L'utilisateur peut librement avoir accès à la puissance de calcul et de stockage des différents serveurs que proposent les fournisseurs de *cloud*.
- **Broad network access** : Les puissances de calcul et de stockage sont disponibles à travers le réseau et sont accessibles via les protocoles classiques utilisés par des clients lourds ou légers (par exemple : smartphones, tablettes, ordinateur, serveur, ...)
- **Resource pooling** : Les ressources du fournisseur d'accès sont regroupés pour desservir plusieurs consommateurs, avec différentes ressources physiques et virtuelles dynamiquement assignées et réassignées selon la demande des consommateurs. Les utilisateurs de *cloud* n'ont aucune connaissance de la localisation exacte de la ressource (espace de stockage, puissance de calcul, ...) qu'ils sont en train d'utiliser, mais ils peuvent spécifier la localisation à un niveau plus élevé (pays, état ou datacenter).
- **Rapid elasticity** : Le *cloud computing* doit être, dans certains cas de façon automatique, capable d'allouer ou de libérer de la ressource en fonction de la demande. Pour l'utilisateur, la capacité du *cloud computing* apparaît souvent comme infinie et peut être accessible n'importe quand et pour n'importe quelle quantité de ressources.
- **Measured service** : Les systèmes de *cloud computing* contrôlent et optimisent automatiquement l'utilisation des ressources en mesurant un certain niveau d'abstraction approprié en fonction du type de service utilisé (par exemple le stockage, le traitement, la bande passante, ...). L'utilisation des ressources peut être surveillée, contrôlée et rapportée, en assurant la transparence tant pour le fournisseur que pour le consommateur du service utilisé.

Les avantages du *cloud computing*, comparés à une architecture traditionnelle, ont été identifiés dans [Armbrust et al., 2010, Höfer and Karagiannis, 2011, Li et al., 2010, Oh et al., 2010, Wang et al., 2010, Zhang et al., 2010] et regroupent : une simplicité d'accès avec des services (infrastructures et applications) accessibles sur simple demande avec une qualité de service, un coût uniquement lié à la consommation des ressources utilisées sans avance de frais, des ressources accessibles via Internet à partir de n'importe quel appareil ce qui assure également une légèreté des applications, et une élasticité des ressources en fonction des besoins grâce à l'utilisation d'une infrastructure évolutive et flexible.

## 2.3 L'architecture du *cloud*

De façon générale, l'architecture du *cloud* est composée de 4 couches décrites par [Zhang et al., 2010] : le matériel informatique (*hardware*), l'infrastructure, la plateforme et l'application (figure 2.2 ).

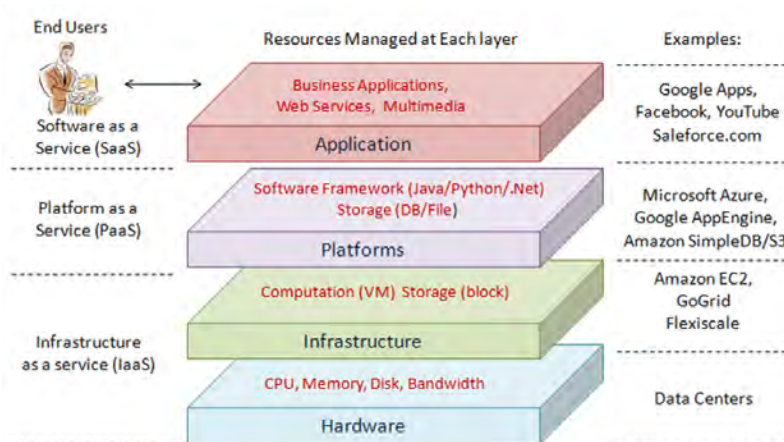


FIGURE 2.2 – L'architecture du *cloud computing* [Zhang et al., 2010]

**Matériel informatique :** Cette couche est responsable de toute la partie physique des ressources du *cloud*, telles que les serveurs, les routeurs, les switches, les systèmes d'alimentation et les systèmes de refroidissement. Les difficultés principalement rencontrées à ce niveau sont la configuration des serveurs, la résistance aux pannes, la gestion du trafic et la gestion des systèmes d'alimentation et de refroidissement.

**Infrastructure :** Également connue comme la virtualisation, cette couche crée des groupes de ressources de stockage et de processeur en partitionnant les ressources physiques par des technologies de virtualisation telles que Xen [Xen, 2003], KVM [KVM, 2008] et VMware [VMware, 1998]. Cette couche d'infrastructure est l'une des couches essentielles du *cloud*, étant donné que certaines caractéristiques comme l'élasticité sont possibles grâce à la virtualisation.

**Plateforme :** Cette couche, mise en place par dessus l'infrastructure, est constituée de systèmes d'exploitation et de *frameworks*. L'objectif de la plateforme est de faciliter le déploiement des

applications directement dans la machine virtuelle. Par exemple, Google App Engine fournit une API (*Application Programming Interface*) afin de gérer le stockage, les données et les utilisateurs d'une application web.

**Application :** La dernière couche de l'architecture est celle de l'application. À la différence des applications traditionnelles, une application *cloud* tire parti de la fonctionnalité automatique de l'élasticité afin d'obtenir de meilleures performances et une meilleure accessibilité à moindre coût.

Cette architecture en couche permet à chacun des niveaux d'évoluer séparément étant donné que chaque couche est indépendante des autres. Ceci permet au *cloud* d'être encore plus modulable qu'une architecture classique.

## 2.4 Le *cloud* au service

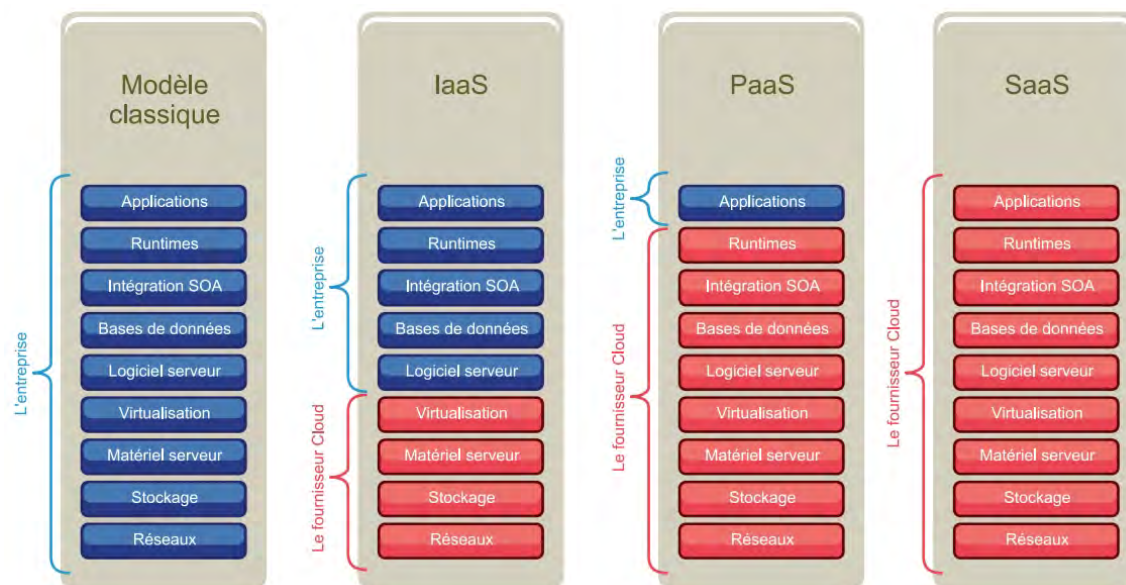


FIGURE 2.3 – Différents types de modèle de services dans le *cloud computing* [Numérique, 2012]

Les fournisseurs de *cloud* s'adaptent totalement aux besoins des utilisateurs et peuvent mettre à disposition différentes parties des architectures machines (figure 2.3).

Dans un modèle classique, l'entreprise a la main sur tout, autant ses applications que son matériel informatique. Dans le cadre du *cloud computing*, la NIST a recensé 3 types de modèle de services où les fournisseurs donnent accès à différentes parties de l'architecture [Mell and Grance, 2011, Bhardwaj et al., 2010, Dillon et al., 2010] : *Infrastructure as a Service* (IaaS), *Platform as a service* (PaaS) et *Software as a Service* (SaaS).

**SaaS :** C'est l'un des services de plus haut niveau. Le fournisseur de *cloud* met à disposition des applications qui tournent sur une infrastructure de *cloud*. L'utilisateur ne les possède pas,

mais paie uniquement leur utilisation. Les applications sont accessibles à partir de clients légers tel que des navigateurs internet, tout comme à partir d'une interface d'un programme. L'utilisateur peut uniquement configurer l'application qu'il utilise, mais il ne contrôle ni ne gère l'infrastructure du *cloud* ou encore les mises à jour de l'application ou de son évolution. Ceci revient au fournisseur de SaaS qui s'assure de rendre l'application accessible à plusieurs utilisateurs à travers le web. En exemple de Saas, nous trouvons SalesForce.com, Facebook, Youtube, les emails de Google, des documents de Google, et ainsi de suite.

**PaaS** : Le fournisseur de *cloud* met à disposition un parc informatique virtualisé où le système d'exploitation et les outils d'infrastructure sont déjà mis en place. Ceci facilite le développement et le déploiement des applications sans le coût et la complexité de l'achat et la gestion d'un parc informatique, en fournissant tout ce qui est nécessaire pour la création et la mise en place d'applications et de services web entièrement accessibles depuis Internet. Le PaaS inclut des *middlewares*, des outils pour développer et pour gérer les données. Contrairement au Saas, l'utilisateur a le contrôle des applications et peut ajouter ses propres outils. En exemple de Saas, nous retrouvons Google App Engine, Microsoft Azure ou Amazon Web Services.

**IaaS** : C'est le service de plus bas niveau. Le fournisseur de *cloud* met à disposition un parc informatique (serveur, stockage et réseau) virtualisé sur lequel l'utilisateur peut installer le(s) système(s) d'exploitation(s) de son choix ainsi que les applications. Ainsi, l'utilisateur n'a pas besoin d'acheter le matériel ni d'en faire la maintenance. C'est une évolution de l'hébergement traditionnel qui ne requiert aucun engagement à long terme et qui met à disposition des utilisateurs des ressources à la demande. Contrairement au PaaS, le fournisseur de IaaS ne s'occupe que du maintien opérationnel du *data center* et les utilisateurs doivent déployer et gérer leurs applications comme dans le cas d'un *data center* personnel. En exemple de IaaS, nous avons le EC2 d'Amazon ou Google Compute Engine.

D'autres types de services dérivés ont également vu le jour comme Data as a Service, Business Process as a Service, Desktop as a Service, plus généralement nommé XaaS où 'X' correspond à virtuellement n'importe quoi [Rimal et al., 2009, Zhou et al., 2010, Zaslavsky et al., 2013].

## 2.5 Les structures de déploiement

Un *cloud* peut être déployé de différentes façons en fonction des différents besoins et contraintes. Le NIST en définit 4 [Mell and Grance, 2011, Zhang et al., 2010, Dillon et al., 2010] :

**Le *cloud* privé** : L'infrastructure du *cloud* est mise à disposition pour une unique organisation qui peut contenir plusieurs utilisateurs. Le *cloud* est détenu, géré et exploité par l'organisation, ce qui a plusieurs avantages : meilleure sécurité du système, meilleur contrôle de la performance et de la fiabilité, une non existence des coûts de transfert de données entre le serveur local et le *cloud* public. Il a cependant la contrainte de la gestion du parc informatique et est limité par la capacité de ce dernier.

**Le *cloud* communautaire** : L'infrastructure du *cloud* est mise à disposition pour une communauté d'utilisateurs de plusieurs organisations qui ont des préoccupations communes (par exemple des exigences de sécurité, de conformité, une mission, ...). Le *cloud* peut être détenu, géré et exploité par une ou plusieurs organisations de la communauté.

**Le *cloud* public** : L'infrastructure du *cloud* est mise à disposition pour le grand public. Le fournisseur est indépendant de l'utilisateur et la gestion peut être faite par une entreprise, une

académie ou le gouvernement, ou une combinaison de ces trois. Le fournisseur définit sa propre politique d'utilisation, en gérant la répartition des traitements sur ses différentes machines ainsi que les différents tarifs d'utilisations. L'avantage pour l'utilisateur est qu'il n'a pas besoin d'investir dans l'achat et la maintenance de l'infrastructure, mais en contre partie, l'utilisateur n'a aucun contrôle sur les données, le réseau et les règles de sécurité.

**Le cloud hybride :** L'infrastructure du *cloud* est une combinaison d'au moins deux infrastructures différentes (privée, communautaire ou publique), qui restent des entités uniques mais qui sont reliées par des technologies propriétaires ou standardisées qui permettent de supporter l'utilisation de données et d'applications sur ce *cloud* partagé. Le *cloud* hybride offre une meilleure flexibilité, notamment avec un contrôle et une sécurité plus fine des données grâce à l'utilisation d'une partie de *cloud* privé (ou hybride) tout en facilitant l'élasticité des ressources en fonction de la demande avec la partie publique du *cloud*. Cependant, ceci demande une séparation minutieuse entre la partie privée et la partie publique du *cloud*.

Le choix du modèle de déploiement va dépendre de :

- **Spécificité et généricité :** la capacité d'une application à s'adapter à d'autres cas d'utilisation (généricité) ou seulement dans le cas pour lequel elle a été créée (spécificité).
- **Criticité :** l'importance de plusieurs aspects tels que les interactions, les dépendances, les risques qui découleront dans le cas d'une panne.
- **Élasticité (ou scalabilité) :** l'évolution de la consommation des ressources en fonction de la demande, autant par rapport au nombre de clients que par rapport au nombre de ressources de stockage ou de calcul nécessaires.

Lorsque le besoin d'élasticité est important, une solution *cloud* est tout à fait adaptée à la situation étant donné que, par définition, le *cloud* a une élasticité rapide. Dans le cas d'une application dont l'évolution de la consommation doit répondre rapidement, il y a le choix entre les différents types de *cloud* : public ou privé. Si la spécificité et la criticité sont deux points importants, on choisira un modèle de *cloud* privé (ou communautaire) afin de s'assurer une certaine sécurité. Si au contraire, ce n'est pas du tout un point important, on choisira plutôt un *cloud* public. Si c'est partiellement important (par exemple uniquement les données), alors on pourra envisager un modèle hybride à la situation en fonction de la grandeur des besoins. Si le besoin d'élasticité, de spécificité et de criticité sont faibles, un modèle de virtualisation qui consiste à créer une version virtuelle (plutôt que réelle) de quelque chose (une machine, un système réseau ...) sera le plus adapté. Dans tous les autres cas, l'application est déployée de façon classique. Tout ceci est résumé dans le schéma (figure 2.4).

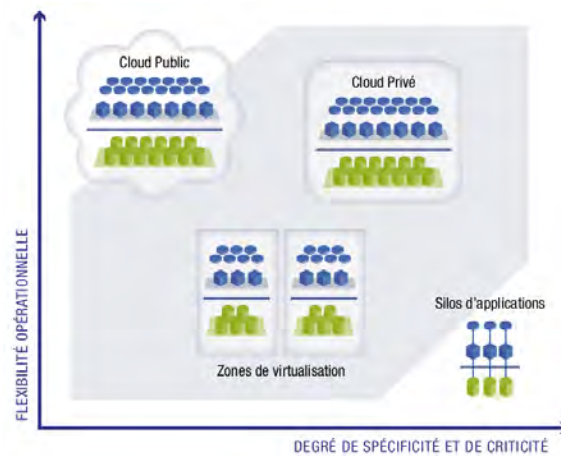


FIGURE 2.4 – Définition de la stratégie de déploiement. Source : Syntec Numérique [Numérique, 2012]



L'analyse des paysages, notamment de l'Arctique, entraîne une quantité de données à traiter de plus en plus importante. Une bonne partie de ces données sont acquises au cours d'expéditions sur le terrain pendant une période donnée. Ceci implique que les traitements ne se feront pas de façon rectiligne à travers le temps mais plutôt de façon ponctuelle et intensive. L'élasticité est donc une caractéristique importante à prendre en compte dans le choix de l'architecture, ce qui nous amène à un choix d'utilisation de *cloud*. De plus, même si certains traitements peuvent être génériques, d'autres peuvent devenir très spécifiques en fonction du type de donnée acquise. De ce fait, les différents types de *cloud* peuvent être intéressants à envisager.

Nous allons nous intéresser à une étude particulière de paysage : celle d'un glacier au Spitsberg où, lors du projet HSF, plusieurs données ont été acquises puis traitées afin d'exploiter plusieurs aspects du glacier.

## Chapitre 3

# Le projet Hydro-Sensor-FLOWS, un glacier sous surveillance

To raise new questions, new possibilities,  
to regard old problems from a new angle,  
requires creative imagination and marks  
real advance in science.

*Albert Einstein*

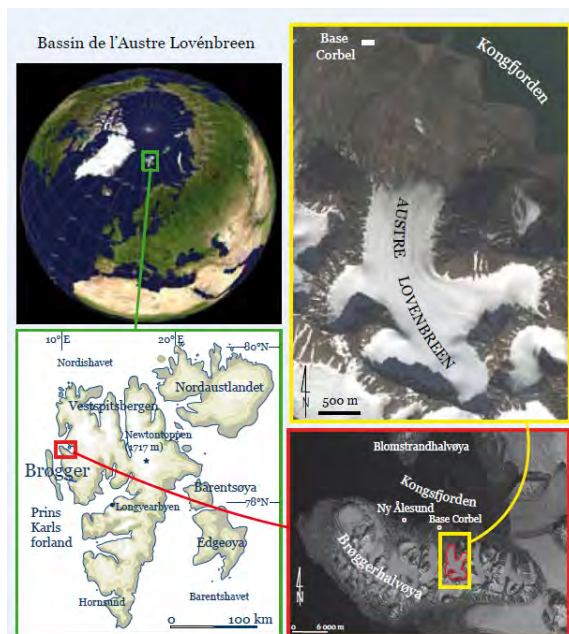


FIGURE 3.1 – Localisation du Spitsberg et du Lovén Est : un terrain d'étude à 1 000 km du Pôle Nord par 79°N. Source : Bernard [Bernard, 2011]

Le glacier étudié par le programme Hydro-Sensor-FLOWS (HSF) est celui du Lovén Est (79°N, 12°E), situé sur la péninsule de Brøgger, à l'ouest du Spitsberg, l'île principale de l'archipel du Svalbard. Le Svalbard est géré par le traité de Paris, qui le désigne notamment sous protectorat norvégien (figure 3.1). Le glacier Lovén Est a été choisi suite à l'historique des données relevées depuis 1953 [Corbel, 1953] par les équipes françaises de géographie. La base scientifique Jean Corbel, créée en 1963, a donné la possibilité à des équipes de scientifiques de venir durant presque 6 décennies, au même endroit, et de permettre des séries de collectes de données comme rarement possibles en Arctique [Corbel, 1966, Griselin, 1982, Péguy and Brosard, 1993, Brosard and Joly, 1994, Griselin et al., 1995, Mercier, 1997].

### 3.1 Le programme Hydro-Sensor-FLOWS

Dans les régions polaires, les glaciers sont des indicateurs paysagers particulièrement visibles des effets du réchauffement climatique sur les paysages polaires [Serreze et al., 2000, Hinzman et al., 2005]. En effet les glaciers sont très sensibles à l'évolution des températures et des précipitations, le retrait des fronts glaciaires se mesure et se voit rapidement (figure 3.2). De ce fait, en Arctique, plusieurs équipes se sont beaucoup intéressées aux glaciers notamment pour mieux comprendre leur évolution, les différents facteurs les influençant et construire des modèles de données permettant de se projeter dans l'avenir [Jania et al., 1996, Dowdeswell et al., 1997, Meier et al., 2003, Gardner and Sharp, 2007, IPCC, 2014].

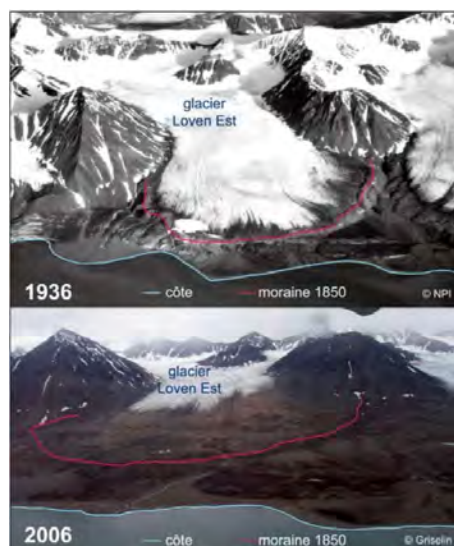


FIGURE 3.2 – Le glacier du Lovén Est est en retrait continu depuis 1936. Les deux clichés aériens illustrent parfaitement ce fort recul. Source : Photo du NPI en 1936, Photo de M. Griselin en 2006

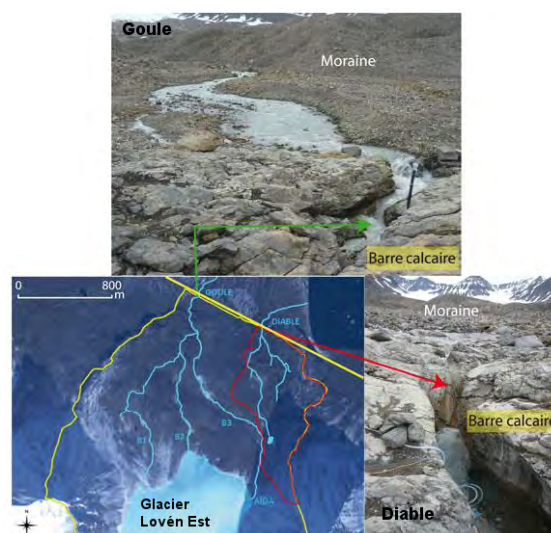


FIGURE 3.3 – Carte et photos des deux exutoires du bassin versant du Lovén Est : la Goule (à l'ouest) et le Diable (à l'est). Adaptation de Quenet [Quenet, 2014]

Lors de la dernière année polaire internationale (2007 - 2009), de nombreux de projets scientifiques ont porté sur la question du retrait glaciaire dans un contexte de changement climatique global. Les recherches sur les variations des volumes glaciaires s'appuient sur les données disponibles telles que les chroniques météorologiques [Griselin et al., 2009a, Marlin et al., 2009], et les photographies aériennes [Corpetti, 2011, Friedt et al., 2011]. Peu de recherches s'appuient sur des données collectées *in situ* notamment parce que cela suppose une présence scientifique de longue date et répétée. La spécificité de la recherche française en géographie au Spistberg est d'avoir procédé à des mesures, des suivis diachroniques d'observations pour le glacier Lovén Est depuis plus de 40 ans à fine échelle [Geoffray, 1968, Kergomard, 1982, André, 1986]. Le suivi comprend le bilan glaciaire du Lovén Est et ses écoulements glaciaires. Le glacier possède des chenaux d'écoulement qui se rassemblent en 2 exutoires avant de sortir de son espace morainique : la goule et le diable (figure 3.3). La concentration des écoulements en 2 points définis dans le temps et l'espace ont facilité les mesures et les études hydrologique sur l'origine des écoulements (précipitations, fonte de la neige, de la glace, du pergélisol) [Griselin et al., 1995, Griselin and Marlin, 1999, Delangle et al., 2010, Marlin

et al., 2013]. C'est sur ce point d'originalité que s'est établi le programme Hydro-Sensor-FLOWS (pour *FLux Of Water and Sediments*).

Le programme Hydro-Sensor-FLOWS (IPY #16, IPEV 304) a démarré en 2006<sup>1</sup>. Le projet rassemble une équipe pluridisciplinaire (hydrologie, géochimie élémentaire et isotopique, géographie – paysage, géomatique et télédétection –, instrumentation RadioFréquences, temps fréquence et capteur *Surface Acoustic Waves*) dirigée par Madeleine Griselin [Griselin et al., 2010]. Ce programme « vise à quantifier, à partir de réseaux de capteurs (en *remote* et *in situ sensing*) et de prélèvements de carottage de neige, les flux liquides et solides d'un hydro-système polaire représentatif des glaciers de type alpin du Spitsberg et suivre la dynamique spatio-temporelle sur quatre ans (2006-2010). Ces observations devaient permettre de comprendre la réactivité aux fluctuations climatiques contemporaines (40 ans). »

### 3.2 Les données, les analyses

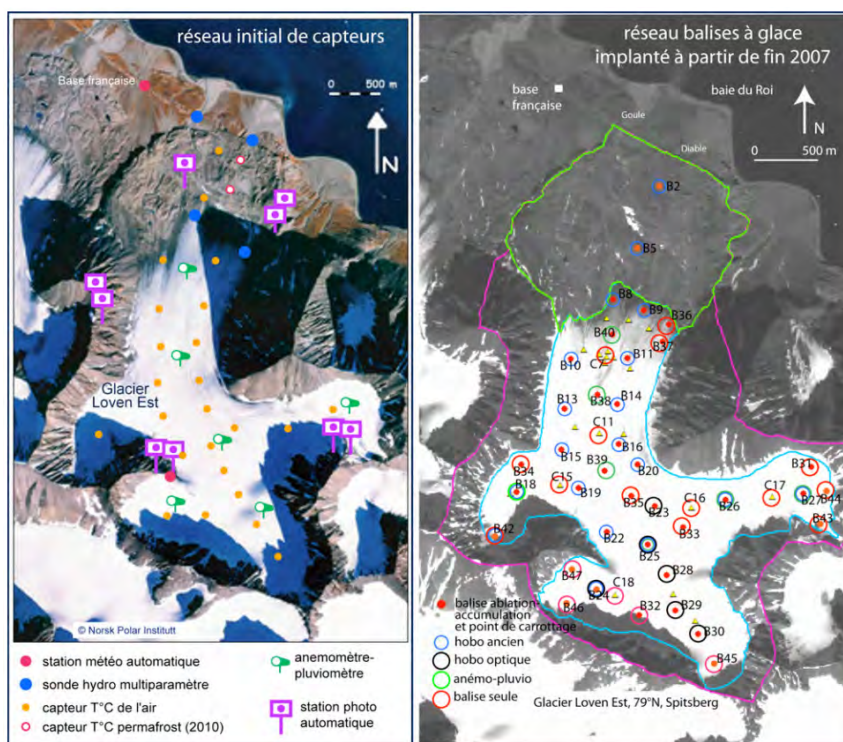


FIGURE 3.4 – À gauche : Installation du réseau initial des appareils de mesures sur le bassin du glacier Lovén Est en 2006. À droite : Évolution du réseau initial, dès l'automne 2007, par l'ajout d'un réseau de balises à glace implantées pour les mesures annuelles d'accumulation-ablation du glacier (bilan de masse) et pour le suivi de la dynamique du manteau neigeux. Source : Griselin [Griselin, 2010]

1. Il repose sur les financements de la région Franche-Comté, de l'Institut Paul Emile Victor (IPEV) et du GDR Mutations Polaires dans le cadre de l'Année Polaire Internationale (2007-2009). Il a obtenu fin 2006 le soutien à long terme de l'ANR (ANR Sensor-FLOWS, 2006-2010).

Le programme HSF se base sur une importante phase de collecte d'informations sur le terrain. Les appareils de mesures installés entre 2006 et 2007 sur le bassin du glacier Lovén Est sont (figure 3.4) :

- une station météo d'altitude
- 22 capteurs de température de l'air (puis stations de mesure de la température du permafrost)
- 7 (puis 9, en 2007) anémomètres-pluviomètres
- 8 (puis 13, en 2007) stations photo automatiques
- 3 stations hydrologiques (sondes multiparamètres et préleveurs automatiques d'eau)

Si les appareils de mesures installés fonctionnent de façon continue et automatique, toute ou une partie de l'année, la maintenance nécessite une intervention physique sur le terrain plusieurs fois dans l'année. À cela s'ajoute des campagnes de terrain pour procéder à des mesures récurrentes (épaisseur et carottage de neige, bilan glaciologique aux balises, positionnement du front, établissement du MNT annuel des altitudes) et aux relevés *in situ* d'ordre hydrologique et hydrochimique des émissaires pendant la saison d'écoulement (jaugeages, prélèvements d'eau à des fins d'analyses hydrochimiques et isotopiques).

### 3.2.1 Changement climatique récent

L'analyse des données météorologiques de Ny Ålesund (1969-2010) a permis de comprendre et quantifier le changement climatique pour la période couvrant les mesures du projet, de 2000 à 2010 [Marlin and Griselin, 2011, Griselin et al., 2009c, Griselin et al., 2009a]. Trois indicateurs du climat ont été retenus : les précipitations (solide et liquide), la température et les vents.

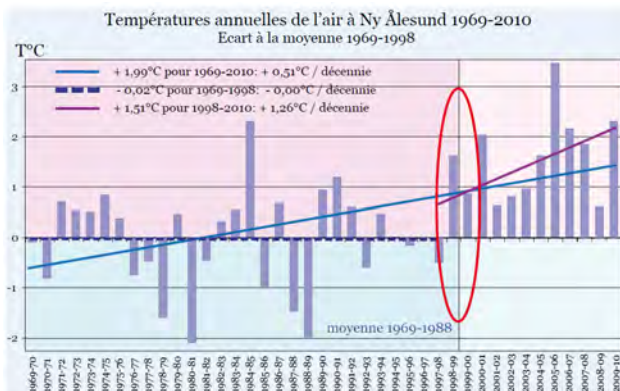


FIGURE 3.5 – Pour la période d'enregistrement 1969-2010 à Ny Ålesund, on observe deux cycles bien différents dont le changement majeur s'est établi pendant les 12 dernières années. Source : Bernard [Bernard, 2011]

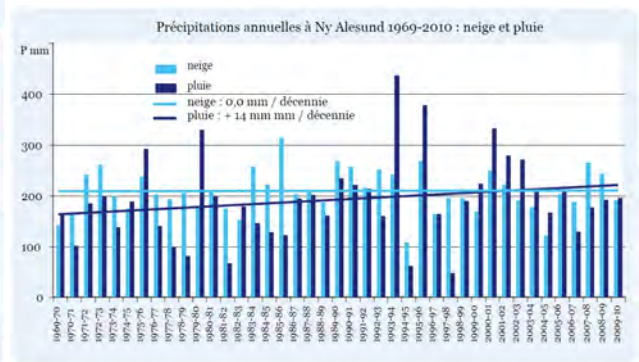


FIGURE 3.6 – Alors que la part de précipitations liquides augmente, les précipitations solides restent très constantes depuis les années 70. Source : Bernard [Bernard, 2011]

La tendance générale pour la presqu'île de Brøgger, relevée par la station de Ny Ålesund sur quatre décennies, montre une augmentation de la température de l'air de 2°C (figure 3.5, trait bleu). Cette augmentation est due à l'augmentation des températures des 12 dernières années (figure 3.5,

trait violet). Il en est de même pour les précipitations dont le volume augmente depuis 30 ans, surtout sous forme de pluie alors que les précipitations neigeuses restent constantes (figure 3.6).

L'analyse des données météorologiques collectées sur le glacier Lovén Est, dans le cadre du programme HSF, précise ces changements à très fine échelle. La présence des 22 capteurs de température sur le glacier a permis, par un calcul d'interpolation, de déterminer l'état thermique du bassin. Cette spatialisation de l'information thermique permet d'attribuer de façon plus fine une valeur de température de l'air. Il est alors possible de déterminer heure par heure la position de l'isotherme 0°C et de quantifier les surfaces de neige et de glace soumises à la fonte et susceptibles d'alimenter les écoulements. Au total, 8 760 cartes d'état thermique sont générées chaque année depuis 2006, ce qui correspond actuellement à la création de plus de 85 000 cartes (figure 3.7).

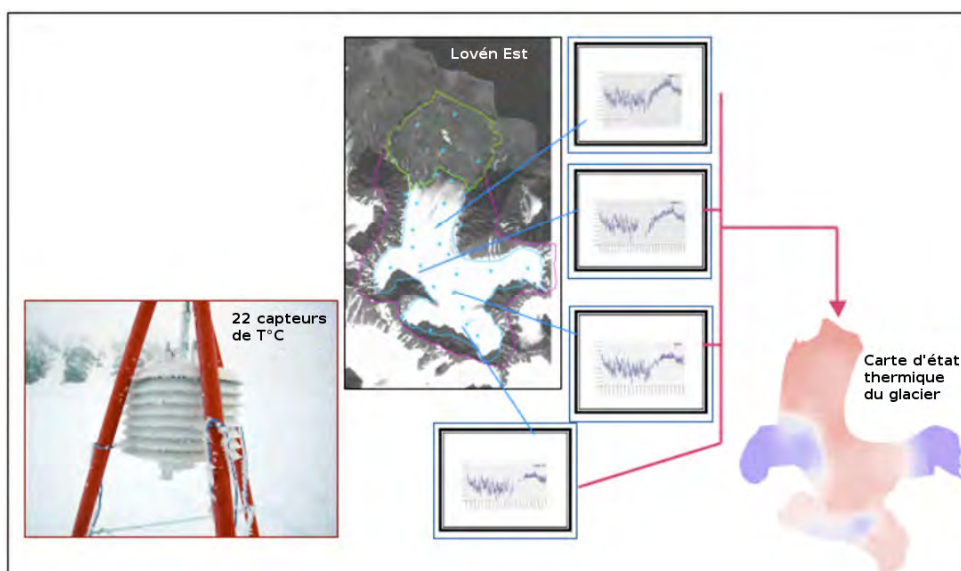


FIGURE 3.7 – Détermination de l'état thermique du glacier. La température est enregistrée heure par heure en 22 points du glacier. Les données sont ensuite interpolées (IDW) pour spatialiser l'information et produire un état thermique du bassin (une valeur unique) à opposer au débit obtenu heure par heure sur les émissaires. Source : Griselin [Griselin, 2010]

L'étude des vents est plus complexe à cause du peu d'anémomètres installés sur le glacier. Les 9 enregistreurs positionnés sur 10 km<sup>2</sup> ne permettent pas une interpolation. Nous avons au moins pu établir les roses des vents en chaque point d'enregistrement, ce qui permet de comprendre les dissymétries spatiales importantes dans la distribution du manteau neigeux (figure 3.8).

### 3.2.2 Dynamique du manteau neigeux et suivi glaciologique du glacier Lovén Est

Pour quantifier la dynamique du manteau neigeux, il est nécessaire de réaliser au printemps, avant le début de la fonte, le suivi des masses et des hauteurs de neige par carottage de neige

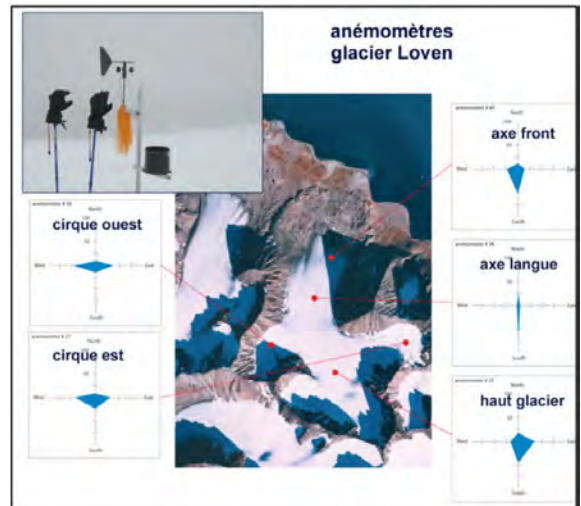


FIGURE 3.8 – Roses des vents sur le glacier Lovén Est. Les cirques ont une dominante est-ouest. Le haut du glacier est soumis aux vents d’est aux vents catabatiques sud-nord. La langue présente exclusivement des vents catabatiques sud-nord, tandis qu’au front, les vents catabatiques sont contrariés par l’influence de la dominante est-ouest affectant le fjord. Source : Griselin [Griselin, 2010]

(drill PICO). Il s’accompagne d’un suivi de la dynamique du manteau neigeux à partir des images satellites Formosat (11 images par an). Ce suivi photographique est complété par les clichés liés au projet HSF. Les appareils de photographie disposés autour du bassin versant<sup>2</sup> du glacier (figure 3.9) permettent de disposer de 3 prises de vue par jour et ce, pour chacun des 10 appareils photo<sup>3</sup>, soit 30 photos par jour, ce qui correspond actuellement à plus de 60 000 photos capturées. L’ensemble de ces photos permet d’affiner au jour le jour l’information concernant la dynamique du manteau neigeux (figure 3.10).

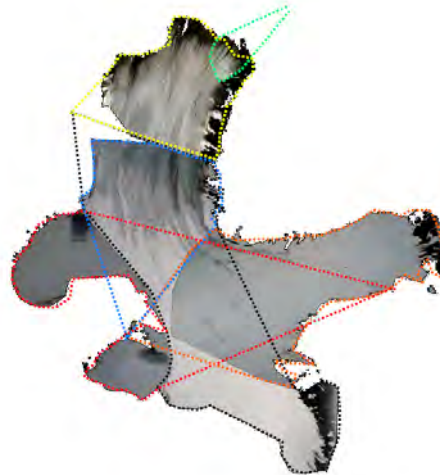


FIGURE 3.9 – Combinaison des photos afin de reconstituer le glacier. L’ensemble des appareils photo couvrent plus de 95% du glacier. Source : Laffly [Laffly et al., 2012a].

2. Un bassin versant est une portion de territoire dont l’ensemble des eaux convergent vers un même point de sortie appelé exutoire : cours d’eau, lac, mer, océan, etc.

3. Ce sont des appareils photo numérique compact du commerce (soit le Leica D-Lux4 soit le Panasonic Luminex LX3).



FIGURE 3.10 – Clichés de la caméra 1, toutes prises le 30 juillet. 2008 montre un retard important dans la fonte puisque le glacier est encore principalement recouvert de neige.

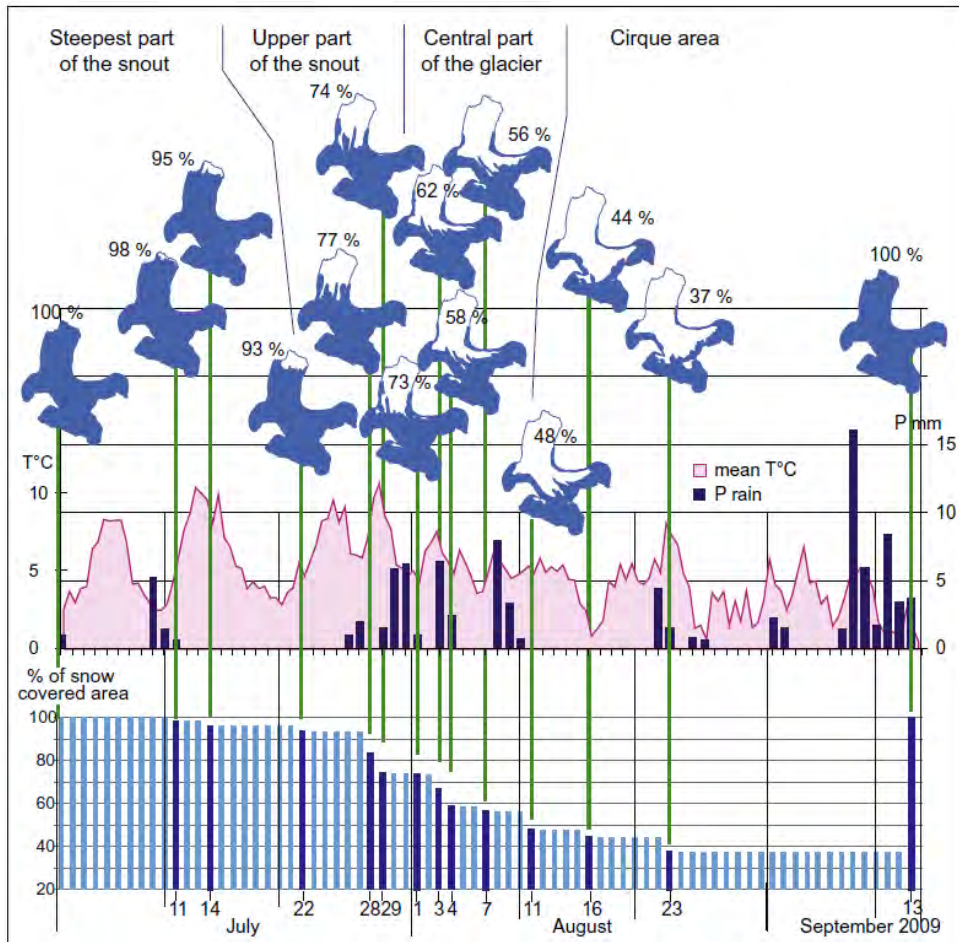


FIGURE 3.11 – En haut : la fonte des neiges au cours d’une saison est séparée en quatre phases principales, avec une évolution de la couverture de neige de 100% à la valeur finale de 37% par rapport à la superficie des glaciers. En bas : la génération de mosaïques permet un calendrier du taux de fusion dans laquelle la variation de la couverture de neige est comparée à des conditions climatiques (précipitations et températures de Ny Ålesund situé à 6 km du Lovén Est ; données fournies par Météo Norsk). Source : Bernard [Bernard et al., 2013b]



Les appareils utilisés ont été sélectionnés pour la qualité de l'optique, la bonne résolution du capteur (10Mpixel) et le stockage sur une carte mémoire SD (*Secure Digital*). Ils sont positionnés directement sur les arêtes du bassin versant du glacier. Les photos acquises par les stations automatiques permettent, une fois orthorectifiées, de faire des calculs très précis, au jour le jour, des surfaces en neige et en glace sur le glacier qui contribuent différemment au bilan hydrologique (figure 3.11). Le travail de redressement géométrique des photos tangentielles est fastidieux (détails au chapitre 5) et peu automatisable (selon les connaissances lors du projet HSF). En plus du suivi de l'évolution de la couverture neigeuse au niveau du glacier, les photos *in situ* permettent également d'étudier la dynamique de la neige sur les pentes (qui constituent 35 % du bassin). En effet, les pentes étant très verticales (environ 35°), leur suivi à partir des images satellites est très limité à cause de la projection verticale qui réduit la zone, et l'accès à pied est trop dangereux (couloirs d'avalanche et éboulis instables) pour y poser des balises ou faire des carottages.

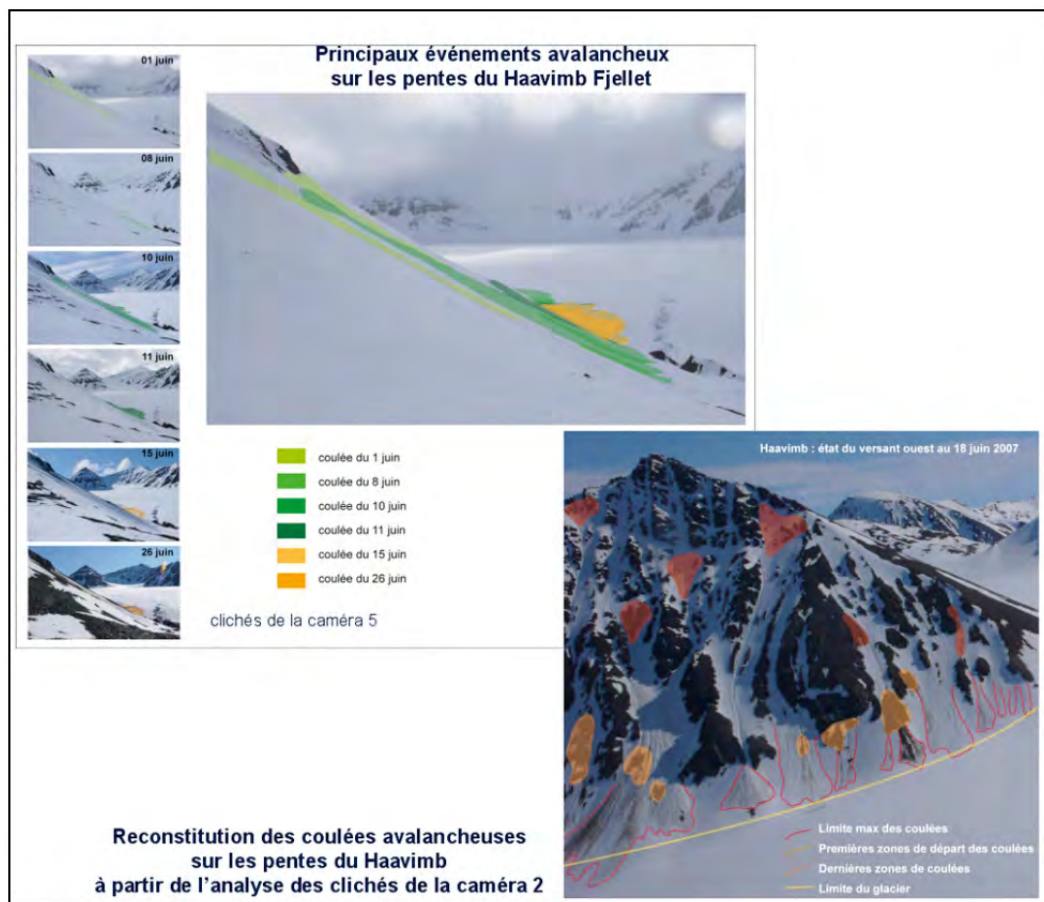


FIGURE 3.12 – Suivi des avalanches à partir des clichés des stations photo automatiques. Les pentes qui représentent 35 % du bassin ne contribuent pas à l'alimentation en neige du glacier, puisqu'aucune coulée n'atteint celui-ci. La neige des pentes est presque exclusivement restituée à l'équation hydrologique par fonte et écoulement direct dans les rimayes latérales. Source : Griselin [Griselin, 2010]

L'exploitation des données a permis d'observer l'impact des avalanches sur l'alimentation en neige du bassin versant du glacier (figure 3.12). Les photos *in situ* ont révélé que la neige accumulée sur les versants mitoyens au glacier n'atteint pas la surface du glacier et ne participe pas à l'alimentation en neige du glacier. Aucune coulée n'atteint le glacier (à l'exception des cirques d'accumulation). La neige accumulée sur les pentes se retrouve intégralement restituée dans le bilan hydrologique par écoulement liquide dans les rimayes latérales où l'eau rejoint les exutoires par circulation latérale sous-glaciaire.

Le suivi de la dynamique du manteau neigeux repose aussi sur des mesures manuelles répétées *in situ*. Les carottages de la neige permettent de définir des hauteurs de neige au point de mesure, d'estimer l'équivalent en eau du volume de neige obtenu par le carottage, à partir des pesées des carottes, exprimé en mm du manteau neigeux. Les méthodes géomatiques permettent d'interpoler les mesures ponctuelles pour spatialiser l'information. L'interprétation des résultats permet d'estimer la lame d'eau équivalente au manteau neigeux, ce qui est stocké ou potentiellement disponible. Lorsque les mesures sont répétées, il est possible de quantifier les variations de volume depuis la précédente mesure indiquant une fonte ou une accumulation (figure 3.13).

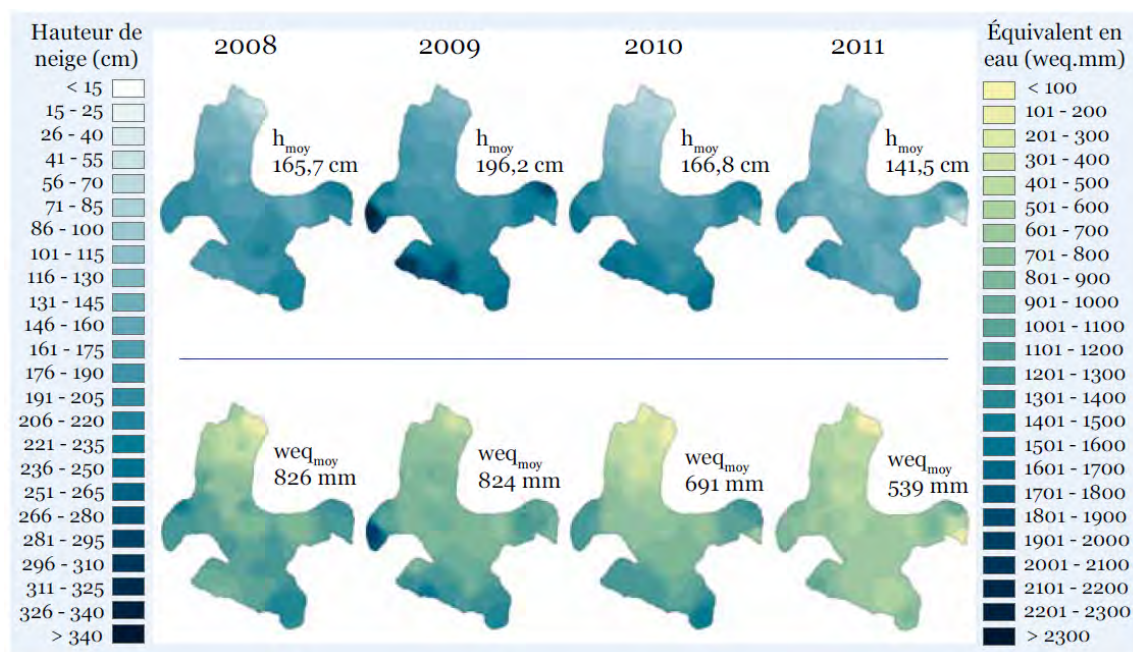


FIGURE 3.13 – Hauteurs de neige et équivalents en eau du manteau neigeux à la fin de l'hiver. En comparant les quatre printemps pour lesquels nous avons effectué des mesures, on constate qu'il y a d'importantes différences tant dans les hauteurs de neige, que dans les équivalents en eau. Des hauteurs de neige similaires peuvent donner des équivalents en eau très différents et inversement. Les hauteurs de neige et équivalents en eau sont représentés par rapport à la surface du glacier. Source : Bernard [Bernard, 2011]

### 3.2.3 Suivi hydrologique du bassin versant du Lovén Est

Le suivi hydrologique du bassin versant du glacier Lovén Est a été réalisé sur deux cours d'eau [Griselin, 2010] :

- La Goule, principal exutoire drainant les écoulements des eaux de fonte du glacier, des versants du bassin et des ruissellements intra morainiques. En été, au cours de la période de fonte, les débits varient beaucoup de quelques litres à quelques  $\text{m}^3$  par seconde.

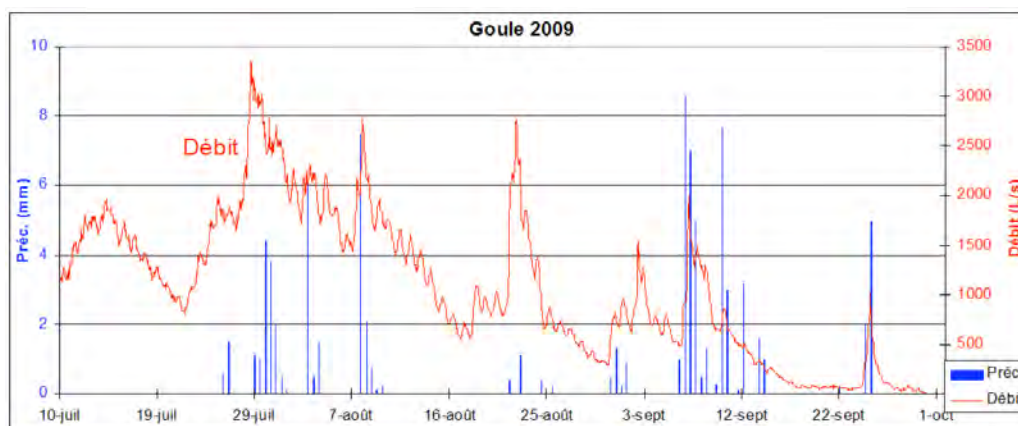


FIGURE 3.14 – Chronique du débit de la Goule (exutoire principal du glacier Lovén Est) durant la saison estivale 2009 et histogramme des précipitations journalières. Source : Griselin [Griselin, 2010]

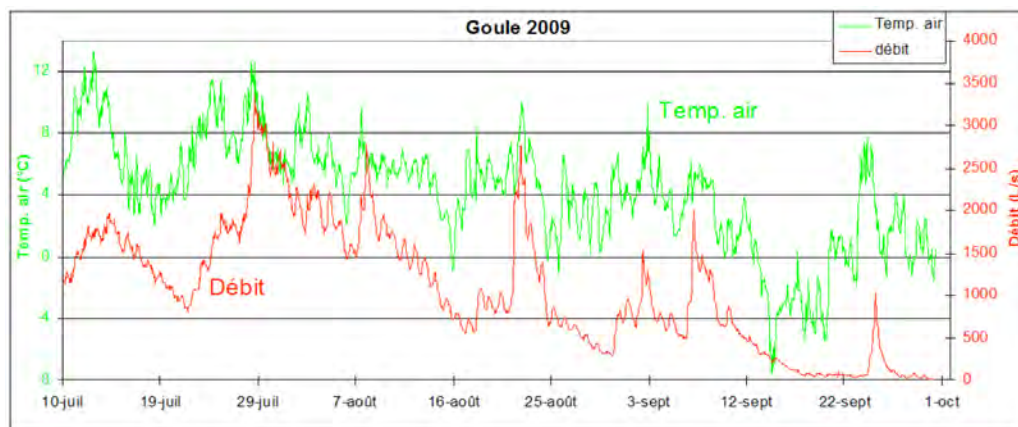


FIGURE 3.15 – Chronique du débit de la Goule (exutoire principal du glacier Lovén Est) durant la saison estivale 2009 et chroniques de la température de l'air. Source : Griselin [Griselin, 2010]

À titre d'exemple, le suivi de 2009 du débit des écoulements à l'exutoire (Goule) superposée aux mesures des précipitations et de la température de l'air (figures 3.14 et 3.15) montre une concordance globale entre la variation du débit des écoulements et les volumes de précipitations. La plupart des événements pluvieux conduisent à des crues événementielles (exemple

du 9 août 2009) mais ils ne contrôlent pas entièrement le débit (exemple du 7 septembre). La température de l'air joue un rôle essentiel dans le comportement hydrologique du bassin versant. Les événements chauds amplifient les écoulements liés à la pluie en libérant des eaux de fonte soit du glacier ou des accumulations neigeuses résiduelles (exemples du 30 juillet et 23 août), pouvant être responsables de crues (exemple du 13 juillet). Enfin, les événements froids agissent instantanément sur le débit, le réduisant au minimum (fin septembre) en arrêtant les écoulements de fonte.

- Le Diable, exutoire secondaire du bassin intramorainique, est déconnecté du glacier bien qu'il puisse y avoir des connections temporaires certaines années ou à certains moments de la saison hydrologique (figures 3.16 et 3.17). Ce petit cours d'eau draine des écoulements de la moraine

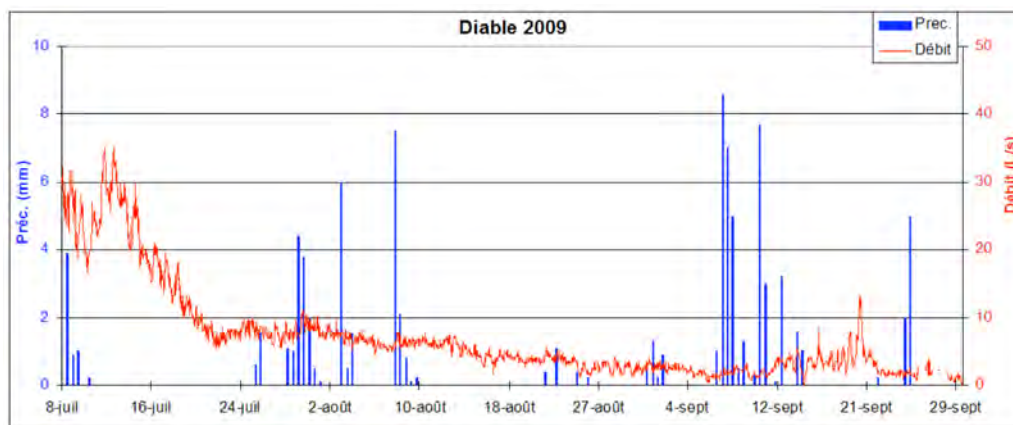


FIGURE 3.16 – Chronique du débit du Diable (exutoire secondaire du glacier Lovén Est) durant la saison estivale 2009 et histogramme des précipitations journalières. Source : Griselin [Griselin, 2010]

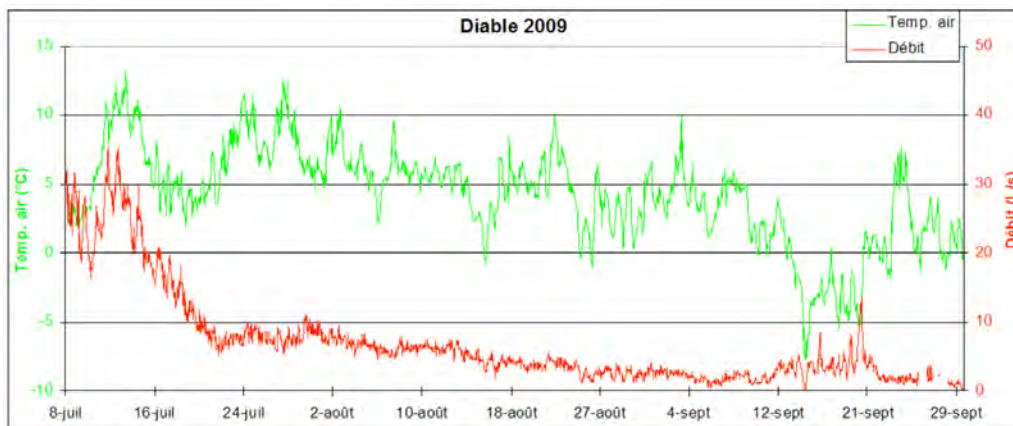


FIGURE 3.17 – Chronique du débit du Diable (exutoire secondaire du glacier Lovén Est) durant la saison estivale 2009 et chroniques de la température de l'air. Source : Griselin [Griselin, 2010]

frontale (neige/pluie, fonte de la couche active) et présente de faibles débits (inférieurs à 50

1.s<sup>-1</sup>).

Dans le cadre du projet HSF, la Goule et le Diable ont été observés pendant plusieurs étés (2007, 2008, 2009 et 2010) [Delangle et al., 2010, Griselin et al., 2009b, Marlin et al., 2009, Marlin et al., 2008, Griselin et al., 2007] à l'aide de capteurs de pression, de sondes multi-paramètres et de préleveurs automatiques d'eau pour les mesures chimiques et isotopiques en laboratoire (figure 3.18).

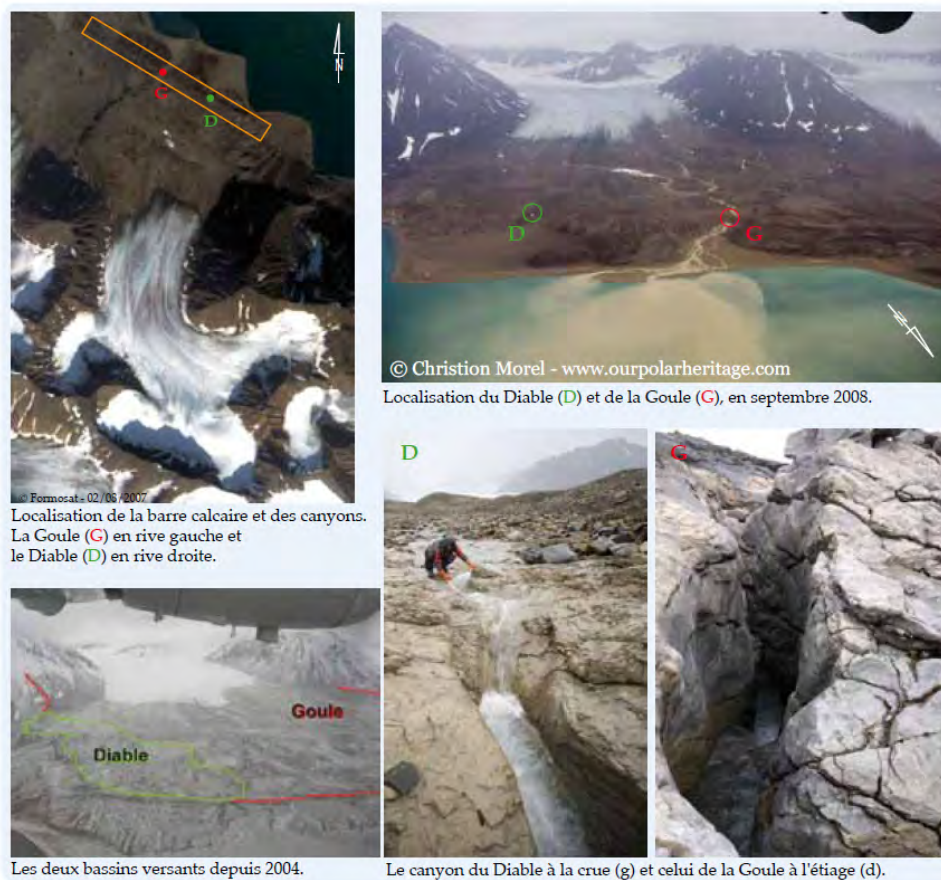


FIGURE 3.18 – Les deux exutoires du Lovén Est, le Diable et la Goule avec leurs bassins versants. Source : Bernard [Bernard, 2011]

### 3.2.4 La géomatique pour quantifier la réaction du glacier aux changements climatiques

Le bilan de masse du glacier (ablation / accumulation), les variations de position du front glaciaire et les calculs du volume du glacier mettent en évidence les réponses du glacier aux variations climatique [Friedt et al., 2011, Bernard et al., 2009, Saintenoy et al., 2011, Saintenoy et al.,

2013, Bernard et al., 2014].

Le bilan de masse, réalisé à la fin de la période de fonte, définit la perte en glace ou le gain en neige du glacier. Il est calculé à partir de la différence entre l'accumulation annuelle mesurée par carottages de neige au printemps et l'ablation globale de la glace du glacier. À partir de 36 points balisés à la surface du glacier, le bilan net est mesuré, puis les valeurs sont interpolées sur la totalité de la surface du glacier afin de donner une représentation spatiale du bilan (figure 3.19). Ce bilan (hauteur par surface) exprime un volume gagné ou perdu par le glacier.

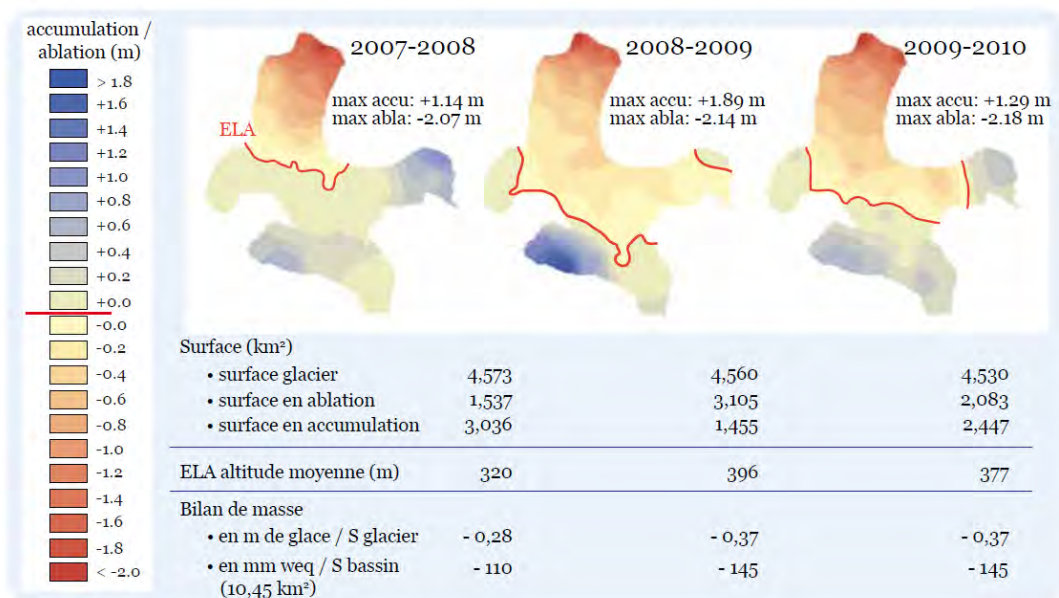


FIGURE 3.19 – Bilans de masse annuels du Lovén Est. Les résultats sont issus des mesures aux 36 balises à glace. L'asymétrie du glacier est particulièrement marquée sur la zone d'ablation. Source : Bernard [Bernard, 2011]



FIGURE 3.20 – Position du front du glacier Lovén Est depuis 2007, mesures effectuées fin septembre chaque année. Source : Griselin [Griselin, 2010]

La ligne d'équilibre moyenne, sur ces trois années, se situe en moyenne à 365 m d'altitude. L'ablation concerne un peu moins de la moitié de la surface du glacier : 34% en 2007-2008, 66% en 2008-2009 et 47% en 2009-2010. Pour autant, les trois saisons présentent des bilans de masse négatifs avec  $-0.28\text{m}$  de glace pour 2007-2008, et  $-0.37\text{m}$  de glace pour 2008-2009 et 2009-2010. Bien que le bilan de masse soit équivalent pour les deux dernières années, les dynamiques sont bien

différentes : en 2008-2009, la très forte accumulation est répartie sur pratiquement un seul cirque, alors qu'en 2009-2010, l'accumulation est beaucoup moins marquée mais elle est répartie sur plus de 50 % du glacier. Ces bilans de masse négatifs impactent la position du front du glacier (figure 3.20).

Le volume du glacier a été calculé à partir de mesures directes sur le terrain avec un GPR (*Ground Penetrating Radar*; figure 3.21). Il a été calculé à  $0,345 \pm 0,0017 \text{ km}^3$ , en tenant compte d'une marge de 5 % d'erreur dans l'estimation de la profondeur [Saintenoy et al., 2011, Friedt et al., 2011]. Rapporté à la surface du glacier, ce volume correspond à une profondeur moyenne de 76 m.

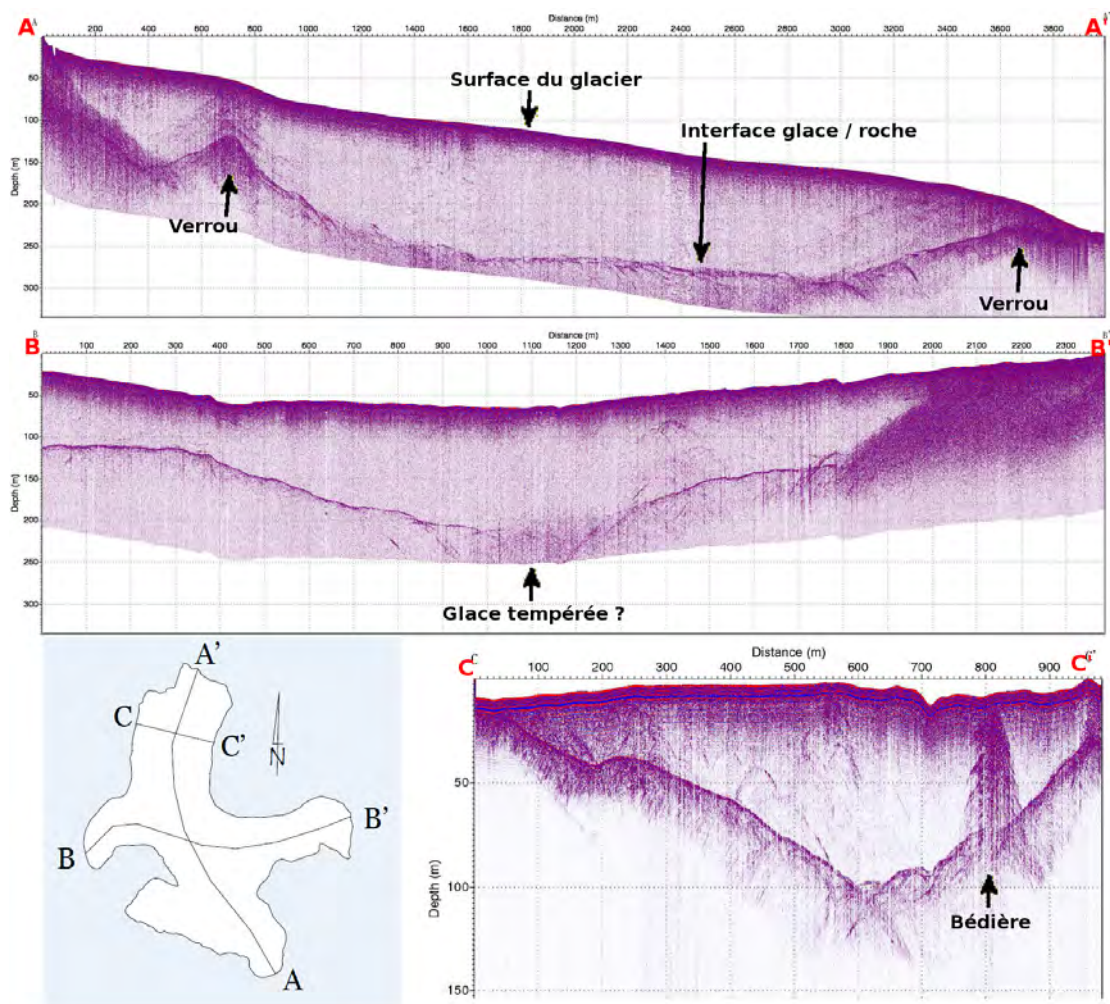


FIGURE 3.21 – Les trois radargrammes illustrent les grands axes du glacier et un exemple de la langue. On distingue clairement l'interface entre la glace et le lit rocheux. Ces mesures permettent ainsi de connaître la topographie sous-glaciaire et intra-glaciaire telle que la bédière matérialisée par la flèche sur le profil CC' (profondeurs exagérées trois fois). Adaptation de Saintenoy [Saintenoy et al., 2011]

Le programme HSF a permis l'auscultation à fine échelle du glacier du Lovén Est, correspondant à une surface de 4,6 km<sup>2</sup>. À l'issue du programme, il en résulte une importante quantité de données acquises sur le terrain à laquelle s'ajoute les données dérivées des différentes analyses. La collecte des données mise en place dans le cadre du programme HSF a été poursuivie par le programme Cryo-Sensor.

Une importante masse de données a été collectée et générée par ces 2 programmes successifs, en 10 ans de mesures. La base de données acquise par le programme HSF compte :

- 300 mesures de bilan de masse interpolées en carte ;
- 800 mesures de densité ;
- 5 000 données de carottage ;
- 3 500 données de température et de précipitation pour Ny Ålesund ;
- 7 000 relevés de précipitation pour le bassin versant ;
- 85 000 relevés de températures interpolés en cartes d'état thermique ;
- 7 000 mesures de lames écoulées, pour la Goule et pour le Diable ;
- 100 images satellites ;
- 60 000 photos *in situ*.

Cependant cette importante masse de données pose un certain nombre de contraintes d'exploitation liées à leur nombre, à leur volume et au matériel informatique nécessaire pour leur traitement. Les techniques de traitements habituellement utilisées par les géographes ne permettent pas l'exploitation de toutes les données collectées dans le projet, notamment lorsque l'on souhaite croiser les différents types de jeux de données. Le *cloud computing* propose des solutions de traitement et d'analyse pour optimiser la gestion de l'hétérogénéité et des volumes de données.





## Chapitre 4

# La variété, l’hétérogénéité et la quantité des données : une base de données *cloud* adaptée

The whole of science is nothing more than refinement of everyday thinking.

---

*Albert Einstein*

L’analyse environnementale collecte et génère une grande quantité de données qui sont variées et hétérogènes. Face à un tel volume d’information, le stockage hiérarchique sous forme de dossiers et fichiers atteint vite sa limite. La manière de stocker les données afin d’assurer une cohérence, d’éviter la redondance, de faciliter le partage des informations tout en assurant la pérennisation prend toute son importance. Les bases de données, non seulement centralisent l’information, mais apportent une alternative au procédé classique de stockage de données.

### 4.1 Base de données structurées

Une Base De Données (BDD) correspond à une collection d’information, de données et de liens entre ces données, qui existent sur une longue période de temps, souvent plusieurs années [Garcia-Molina et al., 2008, Weglarz, 2004]. L’organisation des données est décrite à travers un schéma qui renseigne sur les caractéristiques de chaque type de donnée et les relations entre les différentes données qui se trouvent dans la base de données. Dans le langage courant, la base de données fait référence à une collection de données qui est gérée par un Système de Gestion de Base de Données (SGBD). Un SGBD est un outil puissant pour créer et gérer de grandes quantités de données de manière efficace. Il permet aux données de persister pendant de longues périodes de temps, en toute sécurité. Un SGBD doit pouvoir [Ramakrishnan and Gehrke, 2000, Garcia-Molina et al., 2008] :

- permettre aux utilisateurs de créer de nouvelles BDD et spécifier leurs schémas, c’est-à-dire définir l’organisation des données à stocker et les liens entre ces données de manière logique ;

- assurer l'intégrité des données, en vérifiant que la donnée correspond bien aux contraintes définies par le schéma (par exemple, avant l'insertion d'une date de naissance dans une table, on peut s'assurer qu'elle existe bien et qu'elle n'est pas dans le futur ou dans un passé trop antérieur) ;
- supporter le stockage d'une très grande quantité de données sur une longue période, tout en permettant un accès efficace aux données pour les interroger et les modifier ;
- donner aux utilisateurs la possibilité d'interroger, c'est-à-dire récupérer des informations, et de modifier une donnée présente dans la base ;
- assurer la durabilité du système, ce qui correspond à la récupération des données lors de pannes, d'erreurs en tout genre, ou de mauvaise manipulation ;
- contrôler l'accès aux données par plusieurs utilisateurs en même temps, sans autoriser d'interactions inattendues entre utilisateurs et sans modifier partiellement et non complètement la donnée afin de garder une certaine cohérence dans la BDD.

Afin d'assurer l'intégrité des données, la plupart des SGBD sont basés sur des transactions. Une transaction est une unité de travail, encapsulant généralement un certain nombre d'opérations sur une base de données (par exemple, la lecture, l'écriture ou la suppression de données). Elle garantit la cohérence de la structure des données dans toutes les situations de gestion de données. Ces caractéristiques de fiabilités transactionnelles sont connues sous l'acronyme *Atomicity, Consistency, Isolation, Durability* (ACID) [Haerder and Reuter, 1983] :

- **Atomicité** : assure qu'une transaction se fait au complet ou pas du tout ;
- **Cohérence** : assure que chaque transaction amènera le système d'un état valide à un autre état valide ;
- **Isolation** : doit s'exécuter comme si elle était la seule sur le système ;
- **Durabilité** : assure que lorsqu'une transaction a été confirmée, elle demeure enregistrée même à la suite d'une panne d'électricité, d'une panne d'ordinateur ou d'un autre problème.

Dès les années 70, le développement de l'informatique dans les grandes entreprises implique un besoin d'organiser les données sur des supports électroniques. Le modèle d'organisation des données le plus utilisé depuis cette époque jusqu'à ce jour est le modèle relationnel. Ce modèle est composé d'un ensemble de tables (les relations) dans lesquelles sont placées les données ainsi que les associations. Chaque ligne d'une table est un enregistrement [Codd, 1970, Codd, 1979, Codd, 1982]. L'algèbre relationnelle permet d'effectuer des opérations logiques (par exemple l'union, l'intersection) afin d'en retirer les informations désirées. Elle a été formalisée par le *Structured Query Language* (SQL) qui permet d'exploiter les bases de données relationnelles à travers des SGBD.

## 4.2 Variété et hétérogénéité de la géographie : la limite du relationnel

Parmi les avantages d'une BDD relationnelle, en plus de la centralisation des données, nous retrouvons également la cohérence d'une donnée dans la base à tout instant.

Dans le cadre de l'analyse de paysage, et notamment à travers les différents programmes scientifiques effectués au Spitsberg, les données acquises sont considérables, autant en quantité qu'en

volume. Jusqu'à présent, il n'y a pas de SGBD, à proprement parlé, qui permet de gérer les données, mais chaque chercheur, au sein de sa propre machine ou de son disque dur, définit une structure d'organisation pour ses données (par exemple, par dossiers, par nom de fichiers spécifique), et les traite à sa manière, en fonction de ses habitudes.

L'interdisciplinarité mise en avant suite aux différentes missions de Jean Corbel et des équipes suivantes mène à une génération importante de données souvent structurées mais surtout hétérogènes les unes avec les autres : par exemple, des relevés de températures à une position et une date donnée, des mesures de densité de neige à une position et profondeur donnée, une analyse de la composition de l'eau, des captures d'image de végétation, de glaciers. Les données étant réparties sur chaque ordinateur personnel, il n'y a aucune centralisation ni structure commune entre chacun.

La variété et l'hétérogénéité des données rend la mise en place de schémas, afin de définir une structure logique entre les données, très complexe. Nous arrivons à la limite de la structuration sous forme de relations pour toutes ces données, et comme le souligne Abiteboul [Abiteboul, 1997], l'intégration de plusieurs ressources structurées amène souvent à un modèle de données semi-structuré, parfois appelé non structuré.

### 4.3 Vers un modèle non structuré

Selon Buneman [Buneman, 1997, Buneman et al., 2001], une donnée semi-structurée est souvent décrite comme "sans schéma" ou "s'auto-décrivant". Cela signifie qu'un schéma pré-imposé ou un système de type de données n'est pas nécessaire pour l'interprétation des données semi-structurées, car elles sont généralement représentées sous forme d'arbre ou de graphe étiqueté par des mots clés.

Par exemple, prenons le cas des fichiers Bibtex [Lamport, 1994]. Les données stockées ressemblent à des données suivant un schéma relationnel. Les fichiers Bibtex sont composés d'enregistrements : *article*, *book*, *inproceedings*, ... Mais la structure de ces enregistrements n'est pas régulière. Certains champs peuvent ne pas être renseignés, soit parce qu'ils sont facultatifs soit par omission, et d'autres sont dotés d'une structure significative comme *author*. Certaines fonctionnalités comme les abréviations ou les références croisées ne sont pas facilement exprimables par des schémas relationnels dans les SGBD.

Contrairement aux données structurées dont le schéma définit un certain nombre de contraintes, le schéma des données non structurées est très large afin de prendre en compte l'ensemble des données hétérogènes. Il possède une structure qui peut être irrégulière, implicite et/ou partielle. Le schéma est également évolutif pour pouvoir s'adapter aux différentes évolutions des données [Abiteboul, 1997].

L'évolution des systèmes informatiques et notamment l'arrivée du *cloud computing* répond à une demande de plus en plus grande de stockage et de traitement de l'information. La base de données relationnelle est de moins en moins adaptée à cette situation. Non seulement la quantité d'information augmente, mais ces informations sont devenues de plus en plus variées. C'est ainsi qu'une nouvelle catégorie de SGBD a vu le jour : le *Not Only SQL* (NoSQL).

## 4.4 Bases de données NoSQL, une nouvelle ouverture

Dans le cadre d'une architecture *cloud* qui se doit d'être élastique<sup>1</sup>, les propriétés ACID deviennent un problème [Moniruzzaman and Hossain, 2013]. En effet, l'élasticité gère la montée en charge par l'utilisation de plus de nœuds dans le *cloud* (par exemple, des serveurs ajoutés au réseau initial). Par rapport à une BDD, ceci se répercute par une distribution et/ou une duplication des données sur plusieurs machines. Ce partage de données sur le réseau impacte généralement l'un des trois aspects suivants : *Consistency, Availability, Partition tolerance* (CAP) [Brewer, 2012].

Le CAP théorème affirme que n'importe quelle donnée partagée sur un réseau peut avoir seulement deux des trois aspects en même temps [Gilbert and Lynch, 2012]. Ces aspects sont :

- **C : Consistency (Cohérence)** : signifie que chaque machine renvoie la bonne réponse à chaque demande, comme s'il n'y avait qu'une seule machine ayant la donnée à jour. Cet aspect peut être perdu dans le cas où l'un des nœuds du réseau modifie une donnée et que la mise à jour vers les autres nœuds prend un certain temps.
- **A : Availability (Disponibilité)** : signifie que chaque requête reçoit une réponse. Dans le cas où la cohérence est maintenue, les nœuds devant faire la mise à jour suite à la modification d'une donnée ne sont plus accessibles, rendant donc cet aspect absent.
- **P : Partition tolerance (Tolérance au partitionnement)** : signifie que les nœuds peuvent être partitionnés en plusieurs groupes qui ne communiquent pas entre eux, permettant ainsi de résoudre les problèmes de communication non fiable entre nœuds (perte de messages, panne de systèmes, etc). La cohérence et la disponibilité des données ne sont possibles que s'il n'y a pas de partitionnement, ce qui rendrait impossible ce dernier aspect.

Ainsi, le CAP théorème peut être énoncé comme suit [Gilbert and Lynch, 2012] : « Dans un réseau soumis à des échecs de communications, il est impossible pour n'importe quel service (par exemple, une BDD) présent sur ce réseau de mettre en œuvre une lecture / écriture sur une mémoire partagée qui garantit une réponse à chaque demande. »

Jusqu'à présent, les bases de données relationnelles se sont focalisées sur la cohérence et la disponibilité des données et n'ont pas pris en considération la tolérance au partitionnement. À l'opposé, les bases de données NoSQL ont favorisé la tolérance au partitionnement en renonçant soit à la cohérence, soit à la disponibilité.

La plupart des bases de données NoSQL sont différentes des bases de données relationnelles à cause de leur modèle. On peut définir 4 types de modèles de base de données NoSQL [Hecht and Jablonski, 2011, Moniruzzaman and Hossain, 2013, Cattell, 2011, Zollmann, 2012] :

- **Clé - Valeur** : Le stockage *clé-valeur* est assez similaire au fonctionnement des objets *map*<sup>2</sup> ou *dictionnaire* (exemple en figure 4.1). Les valeurs, souvent opaques au système et stockées dans un tableau d'octets, sont indexées par la clé. La seule façon de venir récupérer les données qui sont stockées dans la base se fait via la clé qui se doit d'être unique pour éviter les ambiguïtés. Les valeurs sont isolées et indépendantes les unes des autres. À cause de cette

---

1. L'élasticité est l'évolution de la consommation des ressources en fonction de la demande, autant par rapport au nombre de clients que par rapport au nombre de ressources de stockage ou de calcul nécessaire

2. Une *map*, aussi appelé dictionnaire ou tableau d'associatif, est un type de données associant à un ensemble de clefs un ensemble correspondant de valeurs. Chaque clef, unique dans l'ensemble des clefs, est associée à une valeur.

simplicité de stockage clé-valeur, ce modèle a une structure entièrement libre. De nouvelles valeurs peuvent être ajoutées en temps réel sans conflit avec d'autres données déjà présentes dans la base et sans modifier l'accessibilité du système. Cependant, le prix de cette simplicité de stockage est la réduction de la flexibilité des requêtes possibles.

Exemples : Dynamo (Amazon) [DeCandia et al., 2007], Voldemort (LinkedIn) [Sumbaly et al., 2012], Redis, Riak.

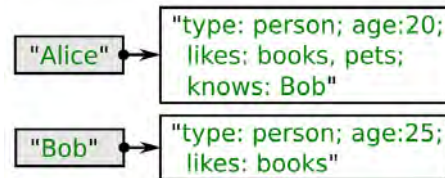


FIGURE 4.1 – Exemple de données représentées dans un stockage *clé-valeur*. La valeur stockée (ici une chaîne de caractères) ne peut typiquement pas être interprétée par le système de stockage. Source : Zollmann [Zollmann, 2012]

- **Document** : Le stockage *document* encapsule les paires clé-valeur dans un document *JavaScript Object Notation* (JSON) ou de structure similaire (exemple en figure 4.2). Dans chacun des documents, les clés des paires clé-valeur doivent être uniques, et les systèmes de base de données documents mettent également en place un identifiant spécifique pour chacun des documents. Contrairement au modèle clé-valeur, les valeurs ne sont pas opaques au système et les requêtes peuvent donc être faites aussi bien sur les clés que sur les valeurs. De ce fait, des structures plus complexes de données telles que des objets imbriqués peuvent être stockées et manipulées. Tout comme pour le modèle clé-valeur, le modèle document n'est pas restreint par un schéma particulier. Le stockage d'un nouveau document contenant autant d'attributs (de paires clé-valeur) de n'importe quelle valeur peut facilement se faire, tout comme la modification de la structure d'un document en temps réel en ajoutant un nouvel attribut. Exemples : CouchDB, MongoDB.



FIGURE 4.2 – Exemple de données représentées dans un stockage *document*. Ici, JSON est utilisé pour formater la donnée. Puisque le format est compris par le système, des requêtes directes sur les attributs (par exemple "name" ou "age") sont possibles. Source : Zollmann [Zollmann, 2012]

- **Orienté Colonnes** : Le stockage *orienté colonnes* vise à regrouper un ensemble de *clé-valeur* par colonne. Il peut également être vu comme un stockage *clé-valeur* avec une clé à deux dimensions : la valeur stockée est référencée par une clé *colonne* et une clé *ligne* (exemple en figure 4.3). En fonction de l'implémentation de la base, le domaine de définition de la clé peut varier. Elle peut également être multidimensionnelle. Ce changement d'orientation donne la

possibilité de rajouter des colonnes sans avoir forcément besoin de venir redimensionner les lignes (les ensembles de *clé-valeur*).

Exemples : Bigtable (Google) [Chang et al., 2008], Cassandra (Facebook) [Lakshman and Malik, 2010].

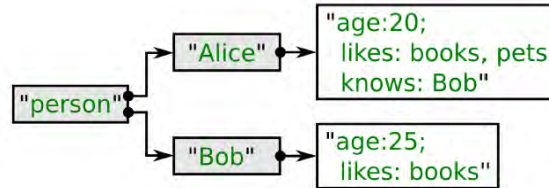


FIGURE 4.3 – Exemple de données représentées dans un stockage *orienté colonnes*. Ici, "person" est utilisé comme la clé de la colonne et chaque nom de personne comme la clé de la ligne. Comme dans le stockage *clé-valeur*, la valeur stockée ne peut pas être interprétée par le système. Source : Zollmann [Zollmann, 2012]

- **Graphe** : Le stockage *graphe* est spécialisé dans l'efficacité de la gestion de données fortement liées (exemple en figure 4.4). Il est le meilleur pour représenter des données inter-connectées, où l'importance de l'association entre les données est équivalente à celle des données elles-mêmes (par exemple, les réseaux sociaux). Ce stockage est basé sur une structure de graphe, et la théorie et les algorithmes sur les graphes peuvent être utilisés lors des différents accès. Exemples : Neo4j [Miller, 2013], GraphDB [Güting, 1994].

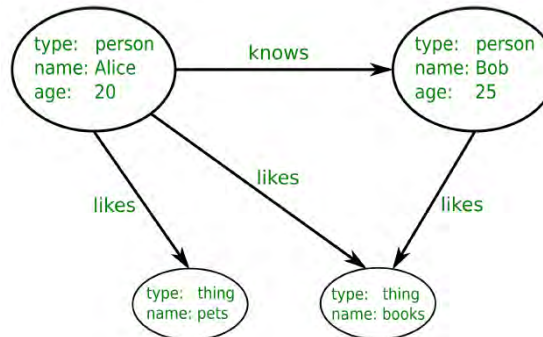


FIGURE 4.4 – Exemple de données représentées dans un stockage *graphe*. Les arêtes peuvent être utilisées pour représenter l'association entre les informations, et les sommets contiennent les informations des objets. Source : Zollmann [Zollmann, 2012]

Parmi ces différents types de BDD NoSQL, le stockage *document* correspond le mieux à nos besoins. En effet, dans le cas de l'étude du glacier du Lovén Est, nous faisons face à plusieurs types de documents dont des photos. Pour chacune de ces données, nous sommes principalement dans un schéma clé-valeur, avec, par exemple, les informations de la date, de la position et de la valeur de la donnée acquise. Pour le cas des photos, elles sont traitées (plus de détails lors du chapitre 5) et le nombre d'images intermédiaires générées varie en fonction des perturbations subies. La structure de stockage doit donc pouvoir être évolutive.

Suite à l'étude menée par Hecht et Jablonski [Hecht and Jablonski, 2011] sur les comparaisons de différentes implémentations de ces différents modèles de bases de données NoSQL, nous avons choisi MongoDB pour sa meilleure gestion de la cohérence des données dans un environnement distribué par rapport à CouchDB.

## 4.5 MongoDB, une base de données adaptées

MongoDB est une BDD NoSQL *documents* libre, introduite par l'entreprise 10gen. Elle a été conçue afin de pouvoir prendre en charge les besoins grandissants de stockage de données [Chodorow, 2011, Zollmann, 2012, Arora and Aggarwal, 2013, Dede et al., 2013].

Les structures de base utilisées par MongoDB sont les collections. Elles sont partiellement équivalentes aux relations (ou tables) du modèle relationnel, mais contrairement aux relations, une collection ne respecte pas un schéma fixe. Elle peut contenir des documents complètement différents. Cependant, donner à tous les documents d'une collection une structure quelque peu similaire peut permettre une interrogation plus facile et plus efficace des données [Zollmann, 2012, MongoDB, 2016].

MongoDB stocke et transmet les documents au format *Binary JSON* (BSON). Tout comme les objets JSON, un document BSON consiste en un ensemble de paires clé-valeur, où chaque valeur peut être un type simple (chaîne, entier), un type complexe au format BSON ou encore une collection.

Comme le postule le CAP théorème, MongoDB garantit la tolérance au partitionnement et, en fonction de la configuration, soit la disponibilité soit la cohérence des données est partielle.

Dans le cas d'un support *cloud*, MongoDB met à disposition un mécanisme appelé *sharding* afin de distribuer les collections à travers plusieurs nœuds de façon transparente pour l'application. C'est une surcouche qui est basée sur le fonctionnement *maître/esclave* ou *replica sets*. Le fonctionnement *maître/esclave* correspond à une répartition des fonctionnalités à travers plusieurs nœuds. Un premier nœud officie en tant que maître et s'occupe des demandes des clients. En parallèle, il s'occupe de répliquer les données sur le nœud esclave de façon asynchrone. L'esclave est présent pour prendre la place du maître si ce dernier venait à être inaccessible. Le fonctionnement *replica sets* correspond à une réplication des données à travers plusieurs nœuds. Un nœud principal, élu parmi et par les différents nœuds du *replica set*, est chargé de communiquer avec le client à travers les différentes actions de lecture et d'écriture. Les nœuds secondaires peuvent être configurés pour qu'ils s'occupent de répondre aux clients en lecture. Tout comme dans le fonctionnement *maître/esclave*, les informations écrites sur le nœud principal sont répliquées sur les nœuds secondaires.

Le *sharding* découpe l'ensemble des nœuds à sa disposition en un ensemble de *shards*, chacun identifié par une clé. Cette clé est utilisée pour découper les collections de données en plusieurs parties appelées *chunk*. Chaque *chunk* correspond à un sous-ensemble de collections, avec ses documents, découpé en fonction de la plage définie par la clé.

La tolérance au partitionnement est clairement définie grâce au *sharding*. Dans le cas de la mise en place du *sharding* par *replica sets*, les données sont toujours disponibles, soit par le nœud principal soit par les nœuds secondaires, mais les données ne sont qu'éventuellement cohérentes en fonction du moment de réplication des données sur les nœuds secondaires. Dans le cas de la mise en place du *sharding* par *maître/esclave*, les données seront toujours cohérentes étant donné que le



client n'a accès que par le nœud maître, mais, en contre partie, la disponibilité est laissée un peu de côté.

Ces fonctionnements de MongoDB la rendent adaptée pour la problématique des données variés, hétérogènes et en grande quantité que génère l'analyse environnementale. Elle est tout particulièrement adaptée pour le stockage des différentes photos ainsi que leurs différents états lors de leurs traitements pour l'analyse de l'évolution de la couverture neigeuse qui correspond à l'aspect du programme HSF que nous allons étudier.

## Chapitre 5

# Analyse de l'évolution de la couverture neigeuse par photo *in situ*

If we knew what it was we were doing, it  
would not be called research, would it ?

---

*Albert Einstein*

L'un des aspects du programme scientifique HSF, continué par le programme Cryo-Sensor, était d'analyser l'évolution de la couverture neigeuse, dans l'objectif de déterminer la vitesse de fonte du manteau neigeux et d'approximer la position de la ligne d'équilibre neige/glace (accumulation/ablation) dans le temps et dans l'espace.

### 5.1 La surveillance du manteau neigeux

Afin de pouvoir cartographier l'évolution du manteau neigeux, il est nécessaire d'avoir une vision globale du glacier de façon régulière afin de pouvoir en déduire son évolution. À cette fin, il a été mis en place sur le terrain 13 stations photo automatiques dont 6 en position fixe, dédiées à l'étude des dynamiques glaciologiques. Utilisant une technique similaire à celles pratiquées dans les Alpes pour le calcul des volumes de glace ou de neige évacuées par un glacier ou lors d'avalanches [Soruco et al., 2011], ou à celles pratiquées en Arctique afin de surveiller les fronts marins ou les dynamiques de manteaux neigeux sur les piémonts [Hinkler et al., 2003, Hinkler et al., 2002, Rolstad et al., 2000, Sund and Eiken, 2010], ces stations permettent d'effectuer une surveillance en continue du bassin glaciaire du Lovén Est en exploitant les données à la fois tangentielles et projetées.

Elles sont une solution quand le suivi à partir des images satellites n'est pas rendu possible. En effet, le programme HSF a fait l'acquisition d'images satellites issues du capteur FORMOSAT-2. Ce satellite d'orbite héliosynchrone assure une acquisition quotidienne au dessus d'une latitude donnée à la même heure solaire, nous garantissant une comparaison pertinente puisque les conditions

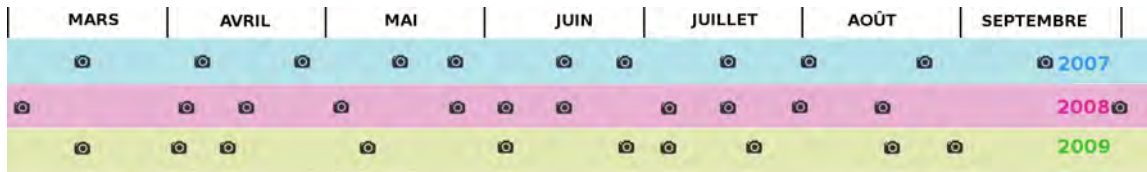


FIGURE 5.1 – Les acquisitions des images FORMOSAT (modélisées par les appareils photo) sont réparties pendant la saison de la fonte. Une comparaison diachronique interannuelle est également possible, puisque les dates d’acquisition sont proches d’une année sur l’autre. À noter que le programme acquérait 10 à 12 images par an, dans des créneaux de dates préalablement choisis, mais la disponibilité des images considérées comme exploitables par Formosat limitait notre choix.

d’éclairage seront toujours similaire. Les images obtenus sont d’une résolution spatiale de 2m, ce qui correspond parfaitement aux besoins pour une étude à échelle locale. Durant les années du programme HSF, 33 images FORMOSAT ont été acquises durant la saison des fontes, correspondant environ à 11 images par année (figure 5.1).

Outre le prix des images satellites, les mauvaises conditions météorologiques ont significativement réduit le nombre d’images disponibles et exploitables, et elles ont également contraint le choix des dates d’acquisition. Malgré cela, l’objectif initial qui était d’obtenir une sélection d’images durant la saison de fonte de façon homogène et de façon à pouvoir les faire correspondre à des événements climatologiques particuliers concordant avec des mesures ou relevés *in situ* a été respecté. Même s’il a été possible d’acquérir des images à fréquence régulière, le laps de temps entre deux clichés reste trop important pour faire une analyse de l’évolution de la couverture neigeuse précise. Si l’on reprend l’exemple de l’analyse faite par Laffly [Laffly et al., 2012b], les deux clichés que nous avons en figure 5.2 sont séparées de 6 semaines : le 15 août 2008 et le 30 septembre 2008. À première vue, on remarque un enneigement de toute la surface du glacier dû à d’importantes chutes de neige pendant les jours précédant la prise de photo, mais ces clichés ne permettent pas de savoir s’il y a eu une grande période de fonte entre temps. Aucune autre image utilisable n’a pu être récupéré du satellite durant cette période à cause de la présence de nuages, mais l’étude hydrologique a montré un pic de précipitations et de température mi-septembre, modifiant grandement la supposition du taux de fonte du glacier et donc de son ablation. Ce type de résultat montre la limite des clichés satellites et la nécessité d’une autre méthode de surveillance de glacier.

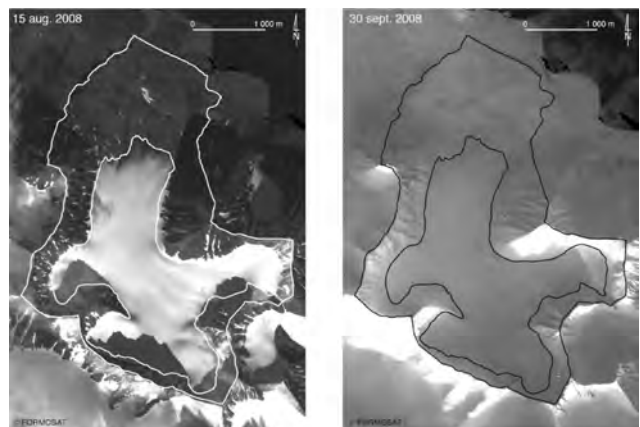


FIGURE 5.2 – Deux clichés satellite FORMOSAT du glacier Lovén Est, acquises en 6 semaines : 15 août 2008 à gauche et 30 septembre 2008 à droite. Ses deux contours : l’interne qui correspond à la partie glace et l’externe qui correspond à la partie géographique du Lovén Est. Source : Laffly [Laffly et al., 2012b].

L'utilisation des photos *in situ* s'est avérée être le seul moyen d'effectuer un suivi régulier. Les stations photo sont placées tout autour du glacier afin d'assurer une couverture maximale du glacier (figure 5.3). Avec les positions choisies, seulement moins de 4% du glacier n'est pas couvert par l'imagerie *in situ* (figure 5.4).

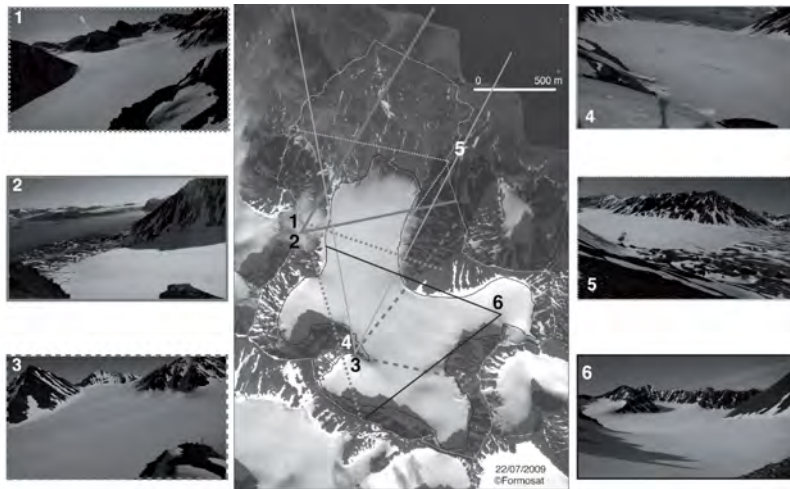


FIGURE 5.3 – La vue de chacune des 6 caméras, ainsi que leur position dans le glacier. Adaptation de Laffly [Laffly et al., 2012b]

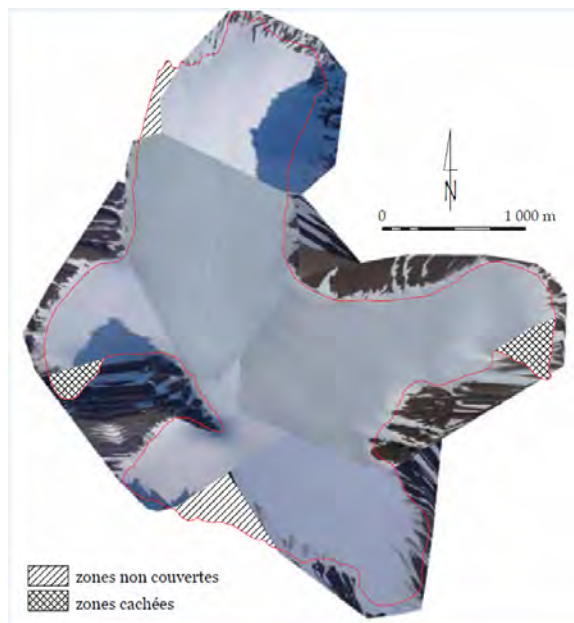


FIGURE 5.4 – La mosaïque construite avec les photos référentielles (celles où le plus de marqueurs étaient visibles) donne une couverture optimale du glacier. Sa surface est couverte à 96% avec quelques zones cachées à cause de la topographie. Les versants présentent un aspect « étiré » dû au choix de favoriser la qualité de la correction géométrique sur le glacier. Source : Bernard [Bernard, 2011]

Les appareils utilisés sont des appareils photo numérique compact du commerce (soit le Leica D-Lux4 soit le Panasonic Luminex LX3) qui ont été sélectionnés pour la qualité de l'optique, la bonne résolution du capteur (10Mpixel) et le stockage sur une carte mémoire SD (*Secure Digital*). Les appareils étant positionnés directement sur les arêtes entourant le glacier, l'autonomie et la gestion

d'énergie sont un problème majeur. Afin de pouvoir fonctionner la majeure partie de l'année, les appareils ne doivent être alimentés que lorsqu'ils sont en fonctionnement pour économiser l'énergie de la batterie. À l'opposé, la pile interne de l'appareil photo permettant d'alimenter l'horloge temps-réel doit fonctionner continuellement afin de garantir d'exactitude des informations dans l'entête EXIF (*Exchangeable image file format*) des clichés. Cette information de la date du cliché est tout autant importante que le cliché lui-même pour le traitement des données. Le froid accélérant la décharge des batteries, des panneaux solaires associés à un accumulateur ont été installés sur les stations photo afin de prolonger la durée de la batterie. La mise en marche des appareils est contrôlée par un interrupteur analogique simulant la manipulation par une personne. Un microprocesseur *Texas Instruments MSP 430* contrôle l'ensemble du processus de capture d'image : configuration de l'heure réelle à l'allumage de l'appareil, mise en route de l'appareil, réglage autofocus, déclenchement de la photo et extinction de l'appareil. Le microprocesseur gère l'horloge en comparant chaque seconde avec l'alarme de déclenchement de cliché pré-réglée par l'utilisateur (figure 5.5).



FIGURE 5.5 – La station photo automatique de 3<sup>e</sup> génération. Ce système, complexe dans sa conception, est adapté aux conditions difficiles de l'Arctique et dispose d'une autonomie qui lui permet un fonctionnement sans intervention humaine pendant la majorité de l'année. Système mis en place par FEMTO-ST, partenaire du programme HSF. Source : Bernard [Bernard, 2011]

Les stations photo ont été programmées pour prendre 3 photos par jour, à 8h, 12h et 16h. Cette fréquence a été choisie afin d'avoir un bon compromis entre la quantité d'information à avoir par rapport au suivi du manteau neigeux et par rapport à l'espace mémoire de chaque caméra. En début et

en fin de saison hydrologique (correspondant à avril et septembre environ), des missions de terrain étaient prévues afin de récupérer les photos des différentes caméras et d'assurer une maintenance du matériel (changement des batteries, vérification de l'étanchéité).

## 5.2 Le traitement des données *in situ*

La mise en place de ce dispositif d'acquisition de photos *in situ* a entraîné deux contraintes techniques principales :

**La quantité de photos :** L'acquisition de 3 photos par jour par 6 caméras pendant les 4 ans du programme HSF a généré une grande base de données (plus de 60 000 photos). Contrairement aux images satellites, ces photos doivent être traitées afin qu'à partir des différentes vues obliques un équivalent de la vue satellite soit créé. Puis, afin de connaître l'évolution du manteau neigeux au cours du temps, un traitement de classification doit être appliqué afin de différencier les états du glacier (neige, glace, roche). Cette quantité de données ne permet pas un traitement manuel. Une première partie de la problématique de cette thèse réside dans l'automatisation des traitements afin de traiter l'ensemble des photos *in situ* acquises.

**Les traitements :** Les traitements à appliquer sont nombreux et différents les uns des autres. Nous pouvons les séparer en deux catégories. Premièrement, les traitements qui permettent de passer les photos en vue cavalière à un équivalent de la vue satellite (projection, mosaïque) et les traitements qui permettent de différencier les différents éléments du glacier (neige, glace, roche) afin de déterminer l'évolution de la couverture neigeuse au cours du temps (classification). Deuxièmement, les traitements qui permettent de réajuster les images en fonction des différentes perturbations subies par les stations photo à cause des conditions extrêmes du milieu polaire (nettoyage, datation, recadrage). Ces traitements sont généralement appliqués avant la transformation en équivalent satellite. On les nommera pré-traitements.

Juste à l'énumération de ces différents types de traitements, nous pouvons remarquer que tous les pré-traitements ne sont pas forcément à appliquer sur toutes les photos. En effet, une photo prise lors d'un beau jour peut avoir à être recadrée, tout comme l'inverse peut être également possible (goutte d'eau sur la lentille ou brouillard, sans pour autant devoir être recadrée), ou encore une combinaison des deux. C'est pourquoi, le choix, la quantité et même l'ordre des pré-traitements à appliquer varie en fonction de la (des) perturbation(s) subie(s). Les traitements principaux (projection, mosaïque, classification) sont à appliquer sur toutes les photos exploitables (parmi toutes les photos acquises, environ 10 000 ont leur date enregistrée dans l'entête EXIF<sup>1</sup>). L'ordre entre les traitements principaux est partiellement important. La mosaïque ne peut s'appliquer que sur des photos déjà projetées, contrairement à la classification des différents éléments du glacier qui pourrait se faire avant la projection, entre la projection et la mosaïque ou après la mosaïque.

De plus, parler d'une classification, d'une projection, d'un nettoyage ou de n'importe quel autre traitement est assez générique. Il existe plusieurs méthodes pour ces traitements, et parmi ces différentes méthodes, certaines peuvent être plus adéquates que d'autres en fonction des contraintes des photos (par exemple, les tons de couleurs très similaires ou la profondeur de la scène). Il existe

---

1. EXIF, *Exchangeable image file format*, est une spécification de format de fichier pour les images utilisées par les appareils photographiques numériques pour stocker des méta-données sur la photo acquise comme la date, l'heure, les réglages de l'appareil, etc

aussi certains traitements qui sont encore dans le domaine de la recherche et dont le résultat obtenu reste partiel, voire inconnu (par exemple le nettoyage).

### 5.2.1 Projection

Les stations photo étant positionnées tout autour du glacier, elles prennent des clichés *in situ* du glacier, et donc en vue cavalière. Notre objectif étant de reconstituer un équivalent de la vue satellite afin de pouvoir cartographier l'évolution de la couverture neigeuse, nous devons transformer notre image initiale, en vue cavalière, vers son équivalent sur un plan : c'est ce qu'on appelle la projection.

La projection est donc une transformation géométrique de l'image. Les photos acquises par les stations correspondent à différentes zones du glacier. L'écart qu'il y a entre le premier plan et le dernier plan des photos se mesure en kilomètres sur le glacier, et en plus de cette profondeur de scène, le relief du glacier ne permet pas d'avoir une déformation "linéaire". Pour ces différentes raisons, un modèle de régression linéaire appliqué sur l'image entière ne permet pas d'obtenir une bonne projection [Hinkler et al., 2002].

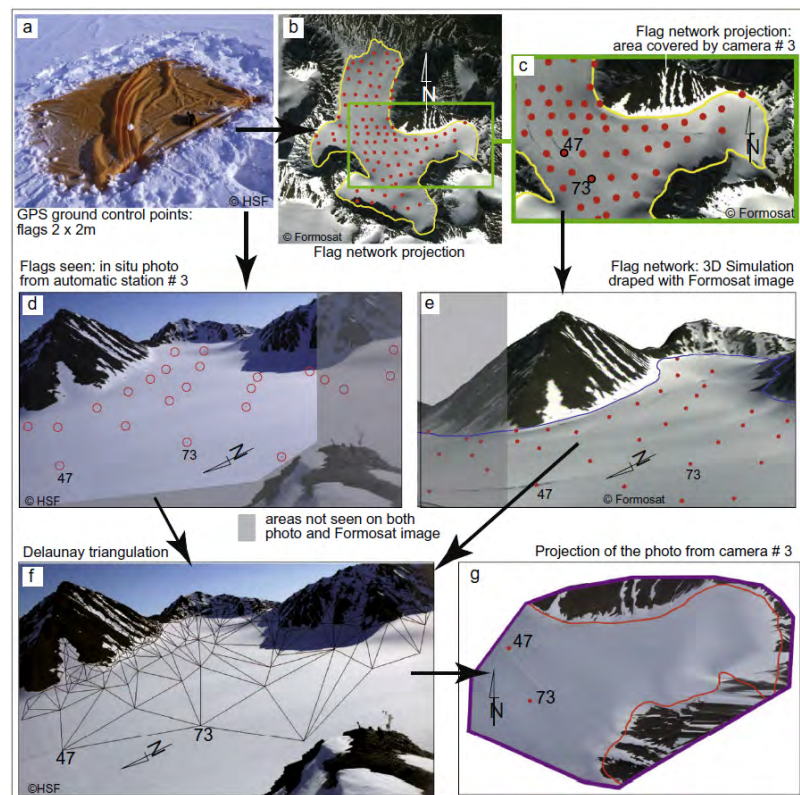


FIGURE 5.6 – Étapes du traitement du signal nécessitant la mesure préliminaire de la position des points de contrôle au sol, sur la surface du glacier. Une problématique importante est l'interprétation des images, afin de localiser ces caractéristiques et identifier quel drapeau est situé à quelle position connue. Source : Bernard [Bernard et al., 2013b]

La méthode de projection adoptée [Laffly et al., 2012b] est celle de *Rubber-Sheeting*. Son principe

est de découper la scène à traiter en un ensemble de zones plus petites, généralement définies par un maillage de Delaunay. Une fois chacune des zones bien définie, le traitement – dans notre cas une régression linéaire – est appliqué localement sur chacune des zones, permettant ainsi un rendu global cohérent [Gillman, 1985, Shimizu and Fuse, 2003].

Afin d’appliquer la méthode de *Rubber-Sheeting* qui utilise un maillage de Delaunay, nous avons besoin d’un certain nombre de points de contrôle ou *Ground Control Points* (GCP) afin de connaître des points équivalents entre la photo *in situ* et le plan, c’est à dire la photo satellite. Il existe déjà un bon nombre de points de contrôle naturels et identifiables à partir des deux clichés (*in situ* et satellite) mais la partie principale qui est celle du glacier n’en contient aucun. Pour pallier ce manque, une centaine de marqueurs géolocalisés (GCP) ont été placés à la surface du glacier. Leur position GPS permet de connaître leur position sur l’image satellite, mais pour en connaître la position sur la photo *in situ*, des drapeaux oranges de 2m × 2m ont été placés à ces mêmes endroits afin de les rendre visibles depuis les stations photo. Pour chaque station, une photo de référence a été prise afin de connaître la position en pixel des GCP, puis chacun des GCP définis sur la photo *in situ* a été relié à sa position dans l’image satellite (figure 5.6).

Le résultat de la projection correspond à une nouvelle image géoréférencée. Le géoréférencement est effectué par un « world file » (fichier de configuration supplémentaire à l’image) qui est constitué de six paramètres qui permettent d’effectuer quatre transformations géométriques, soit : rotation, mise à l’échelle, translation et retournement. La dimension de ces images projetées change en fonction de la prise de vue de chaque station photo. En effet, chaque station capture une vue différente du glacier, avec une profondeur et une largeur de glacier capturées différentes en fonction de sa position (figure 5.3).

## 5.2.2 Mosaïque

Les photos projetées d’une station photo ne nous permettent d’obtenir qu’une partie du glacier. C’est grâce aux six différentes stations que la couverture du glacier est quasi totale. La reconstruction de l’équivalent de la vue satellite nécessite donc la mise en commun des photos projetées provenant des six stations photo : c’est ce que nous appelons la mosaïque.

Étant donné que la mosaïque correspond à un équivalent de la vue satellite, il est important que les six photos projetées soient de la même date afin de garder une cohérence sur les informations du glacier. Comme nous l’avons déjà fait remarqué précédemment, les stations photo sont en autonomie sur le terrain. Les conditions extrêmes auxquelles elles sont soumises ne leur permettent pas toujours d’acquérir une image utilisable à chaque mise en marche des appareils photo. Ceci implique que l’image mosaïque obtenue ne sera pas toujours totale.

L’information de géoréférencement des images projetées est très utile pour la création de la mosaïque, étant donné que chaque image est située de façon universelle et donc est placée directement au bon emplacement sur l’image mosaïque. Cependant, certaines zones du glacier (et notamment la zone centrale) sont présentes sur plusieurs photos projetées. La problématique de ce traitement est la suivante : pour ces zones de l’image mosaïque où plusieurs photos projetées ont de l’information, quelle photo ou partie de photo utiliser pour la mosaïque ?

Deux pistes ont été essayées, apportant des résultats équivalents :

**Utilisation de *shapes* :** Un *shape* correspond à un fichier de configuration contenant une forme



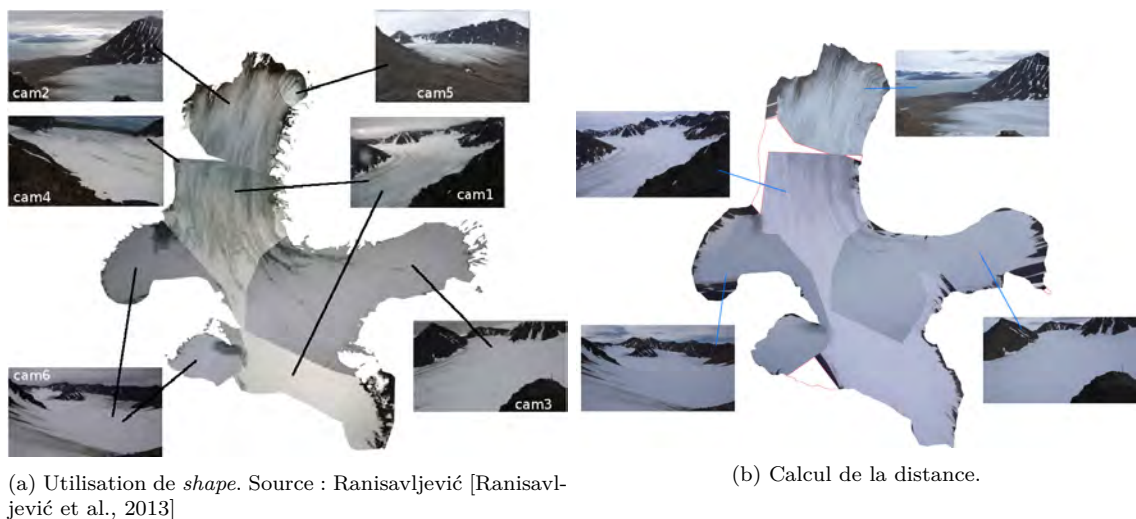


FIGURE 5.7 – Reconstitution d'équivalent de la vue satellite via les images *in situ*.

particulière. Dans notre cas, nous avons défini pour chaque station photo un *shape* correspondant à une zone dans la mosaïque. Ceci permet, lors de la construction de la mosaïque, de savoir, au niveau des zones de conflit, quelle photo choisir (figure 5.7a). L'inconvénient de cette méthode est que la définition d'un *shape* est laborieuse. Et dans les cas où les six stations photo n'ont pas toutes fonctionnées, il est possible qu'une zone qui était précédemment en conflit ne le devienne plus. Ceci sous-entendrait de devoir définir  $x$  *shapes* en fonction de tous les cas de figures, ce qui rendrait le traitement plus complexe.

**Calcul de distance :** Comme mentionné précédemment, la profondeur capturée par les différentes caméras se mesure en kilomètres. Ceci implique que les informations en premier plan sur la photo d'origine sont meilleures que celles du dernier plan, étant donné que la projection déformera plus le dernier plan que le premier. Les pixels de l'image associées au premier plan correspondent à ceux qui sont le plus bas dans l'image, et plus on monte dans la photo *in situ*, plus on recule dans les plans. Par un calcul de positions, nous pouvons définir si une partie d'une image projetée provenant de la caméra  $a$  se situait plus bas (ou plus haut) dans la photo d'origine qu'une image projetée provenant de la caméra  $b$ . Étant donné que nous considérons que plus la position est basse, meilleure est l'information, nous pouvons en déduire quelle information conserver pour la création de la mosaïque (figure 5.7b).

Dans les deux cas, nous n'avons pas pris en compte le modèle d'élévation du terrain, qui pourrait donner encore plus de précisions sur les zones à conserver.

### 5.2.3 Classification

Afin de connaître l'évolution de la couverture neigeuse au cours du temps, il faut pouvoir déterminer les différents éléments visibles du glacier à partir de l'image satellite ou de la mosaïque : neige, glace, roche, ... C'est ce qu'on appelle la classification.

La délimitation entre la neige et la glace se fait, pour le moment, manuellement sur la mosaïque (figure 5.8). Il existe deux grandes familles de classification, appelées *supervisée* et *non supervisée*.

**Classification supervisée :** est une technique d'apprentissage automatique où l'on cherche à produire automatiquement des règles à partir d'une base de données d'apprentissage contenant des "exemples" (en général des cas déjà traités et validés). Plusieurs méthodes de classification supervisée ont été mises en place : *Support Vector Machine* (SVM), le *k-mean*, les réseaux de neurones, etc [Cortes and Vapnik, 1995, Marroquin and Girosi, 1993, Dreiseitl and Ohno-Machado, 2002, Duda et al., 2000, Ripley, 2007, Bishop, 1995, Freund et al., 1999].

**Classification non supervisée :** est une méthode d'apprentissage automatique. Elle fonctionne en divisant un groupe hétérogène de données en sous-groupes, de manière à ce que les données considérées comme les plus similaires soient associées au sein d'un groupe homogène et qu'à l'opposé, les données considérées comme différentes se retrouvent dans d'autres groupes distincts. L'objectif est de permettre une extraction de connaissance organisée à partir de ces données. Plusieurs méthodes de classification non supervisée ont été mises en place : le *fuzzy clustering*, le maximum de vraisemblance et autres méthodes statistiques [Gath and Geva, 1989, Perlovsky and McManus, 1991, Bezdek et al., 1981, Gustafson and Kessel, 1978].



FIGURE 5.8 – Classification du glacier : neige / glace. Source : Laffly [Laffly et al., 2012a]

La difficulté que nous avons face, à l'utilisation de ces méthodes de classification, est l'image sur laquelle nous allons appliquer le traitement. En effet, l'image mosaïque générée correspond bien à un équivalent de la vue satellite, mais, à cause des différences d'exposition solaire, les appareils photo ne captent pas la même intensité lumineuse ce qui fait que la "couleur" de la neige ou de la glace varie d'une photo à l'autre. Cette variation perturbe la classification.

Au lieu de travailler sur la mosaïque, une autre possibilité serait de travailler sur la photo *in situ* d'origine. Malheureusement, suite à certains tests, nous nous sommes aperçus que les méthodes de

classification ne classaient pas le glacier (neige / glace) mais surtout tous les éléments présents sur les photos *in situ* entourant le glacier : le ciel, les montagnes, les nuages, etc.

Finalement, nous avons utilisé l'information de la limite du glacier (donné par un *shape*) afin de ne se focaliser que sur la partie concernant le glacier. Nous avons donc pré-traité les données afin d'extraire uniquement l'information souhaité pour chacune des photos des 6 caméras. Avec ces nouvelles données, nous avons appliqué des méthodes de classification, en prenant en compte non seulement les éléments neige, glace et roche, mais en ajoutant la possibilité d'ombre sur le glacier, et donc de neige ombragée et de glace ombragée. À partir de ces nouvelles données, nous avons obtenu une classification qui se rapproche au mieux de ce qui était souhaité (comparé aux résultats précédents). Cependant, les perturbations météorologiques (brouillard, gouttes d'eau, etc) restent un problème pour la classification.

#### 5.2.4 Nettoyage et utilisabilité

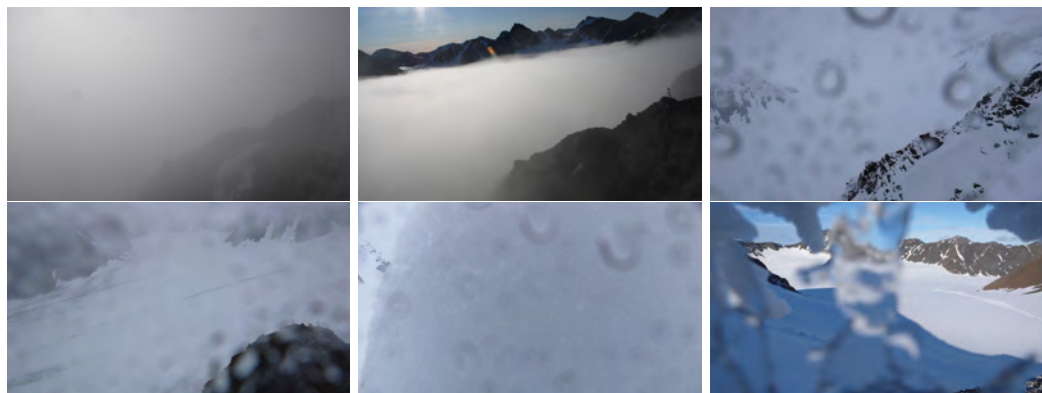


FIGURE 5.9 – Différents types de perturbations : brouillard, mer de nuages, gouttes d'eau sur la lentille, neige sur la lentille, bloc de neige gênant la visibilité, neige et givre partiellement fondus devant la lentille.

L'un des intérêts de mettre en place des stations photo était de pouvoir avoir un suivi plus régulier que les images satellite avec l'avantage que les stations se situent en moyenne altitude et donc peuvent se retrouver sous la couverture nuageuse qui bloque la visibilité du satellite. Malgré cet avantage, étant donné que les appareils photo sont autonomes, ils subissent toutes sortes de perturbations météorologiques de la région telles que les précipitations solide et liquide, le brouillard, le givre (figure 5.9). Certaines perturbations rendent la vue du glacier partiellement percevable, mais dans d'autre cas, rien n'est visible.

La problématique liée à ce pré-traitement est double : comment déterminer qu'une photo *in situ* est totalement utilisable, partiellement utilisable ou inutilisable? Et dans le cas de "partiellement utilisable", existe-t-il un moyen de venir nettoyer l'image en enlevant la perturbation (goutte d'eau, brouillard ...)?

Définir la limite entre partiellement utilisable et inutilisable est un traitement très complexe. Il en est de même pour le nettoyage d'une image. Il n'existe actuellement aucune méthode totalement aboutie sur ce sujet. Cette problématique est toujours un verrou au niveau de la recherche sur le traitement d'images.

### 5.2.5 Datation

Les appareils sont installés tout autour du glacier et sont laissés en autonomie pendant plusieurs mois. Un autocontrôleur permet de déclencher la prise de la photo en laissant 20 secondes à l'appareil pour s'allumer, faire la mise au point, prendre la photo, l'enregistrer sur la carte mémoire et ensuite l'appareil s'éteint afin d'économiser au maximum la batterie (figure 5.10).



FIGURE 5.10 – Caméra dans sa boîte, avec les panneaux solaire afin d'améliorer la longévité de la batterie de la caméra. Extrait de Laffly [Laffly et al., 2012b]

Nous avons remarqué plusieurs perturbations. Dans un déroulement classique du fonctionnement des stations photo, trois photos sont acquises par jour. Or, certains clichés sont inexistant car les perturbations météorologiques rendent la mise au point plus lente voir impossible, ne permettant pas à l'appareil, en 20 secondes, de prendre la photo. La quantité de clichés manquants varie en fonction des situations météorologiques. De même, avant la mise en place de ces panneaux solaires et dans certaines configurations particulières, la batterie des appareils se décharge totalement, rendant inopérant l'appareil photo. Jusqu'à la maintenance suivante, aucun cliché n'est acquis. De plus, les microprocesseurs utilisés pour le programme HSF sont les mêmes que ceux utilisés pour surveiller la mer de glace à Chamonix, mais avec une reprogrammation de la fréquence d'acquisition des photos : au lieu de prendre 6 photos par jour toutes les 2h à partir de 8h, ils prennent 3 photos par jour toutes les 4h à partir de 8h. Dans les situations très particulières où les vents polaires viennent percuter les microprocesseurs, une perturbation électromagnétique est générée, réinitialisant la programmation à 6 photos par jour toutes les 2h. Tant que la pile interne de l'appareil qui permet d'alimenter l'horloge temps-réel n'est pas déchargée, ces conséquences sont gênantes dans le suivi de l'évolution mais il est toujours possible de placer chacun des clichés au cours du temps. Or, le froid qui accélère la consommation d'énergie décharge tout autant la batterie que la pile interne des appareils. Les panneaux solaires permettent d'alimenter la batterie mais la pile interne est inaccessible, rendant sa ré-alimentation impossible. Une fois cette pile interne totalement déchargée, les clichés sont toujours acquis mais l'information du moment où a été pris la photo n'apparaît plus dans le fichier EXIF. Couplé aux perturbations décrites précédemment, la seule information restante est l'ordre dans lequel les clichés ont été pris, mais aucune information ne permet de connaître les dates d'acquisition des photos.

La date étant importante afin de créer une mosaïque d’images du même moment afin de faire une bonne classification et par la suite une bonne analyse, il faudrait trouver un moyen de calculer la date de la photo à partir de l’image.

Actuellement, nous n’avons que quelques pistes associées à cette contrainte mais pas de solution certaine. La piste principale est de travailler avec l’ombre portée des différentes montagnes sur le glacier (figure 5.11). En effet, pour une position donnée du soleil, celui-ci génère un type d’ombre en fonction du modèle d’élévation du terrain. Donc avec la connaissance du modèle d’élévation du terrain (à quelques approximations près) et la position des ombres des montagnes sur le glacier, faire le calcul inverse permettrait de retrouver la position du soleil et d’en déduire la date (et l’heure) associée. Malheureusement, déterminer l’ombre sur le glacier n’est pas une tâche si facile, étant donné qu’il est assez rare de se trouver dans le cas où il n’y a aucun nuage dans le ciel et donc que l’ombre visible sur le glacier ne soit pas perturbée par la présence de nuages. De plus, la quantité de neige vient faire varier le modèle d’élévation du terrain de plusieurs mètres, ce qui vient rajouter des erreurs au niveau du calcul de la date et de l’heure. Même si nous arrivions à déterminer la position de l’ombre et recalculer une date, cette date ne serait que approximative et il est difficile de déterminer la fourchette dans laquelle la date pourrait être.



FIGURE 5.11 – Ombres portées pour une caméra donnée. Source : Laffly [Laffly et al., 2012b]

### 5.2.6 Recadrage

À cause de ces différentes perturbations météorologiques qui ont des impacts variables sur le matériel mis en place, il est nécessaire de faire de la maintenance sur le matériel (figure 5.12). À chaque début (vers avril) et fin (vers septembre) de saison, des missions ont été planifiées. En fonction du type de perturbation, il est parfois nécessaire que l’appareil photo soit enlevé de son pied, réparé puis réinstallé sur son pied. Lors de la réinstallation, le cadrage est fait manuellement, de façon approximative, ce qui peut provoquer un décalage faible ou important en fonction des moments (figure 5.13). De plus, au cours du temps, entre les rafales de vent qui peuvent faire légèrement bouger les appareils qui sont plus exposés à la prise de vent et les autres types de variations externes (par exemple, les animaux de la région), le cadrage peut également changer entre deux interventions de maintenance.

Comme vu précédemment, le traitement de projection nécessite des GCP qui ont été définis en 2009. Cette année est devenue celle de référence, et si on veut que la projection soit exacte, il faut que toutes les photos soient cadrées comme celles de références.

Les déplacements que peuvent subir les appareils sont uniquement des rotations, car, étant donné que le pied sur lequel ils sont placés est fixé au sol, ils ne pourront jamais faire un déplacement horizontal ou vertical. Une régression linéaire d’ordre 2 est donc adaptée à ce traitement de recadrage. Ceci nécessite d’avoir un certain nombre de points de contrôle afin de réaliser le recadrage.



FIGURE 5.12 – Le climat extrême a parfois des conséquences importantes et inattendues sur les stations photo. Du dos arraché au saphir enfoncé (a et b), de la gangue de glace à la station remplie d'eau (c et d), nombreux sont les aléas qui peuvent perturber le bon fonctionnement des appareils photo. Source : Bernard [Bernard, 2011]

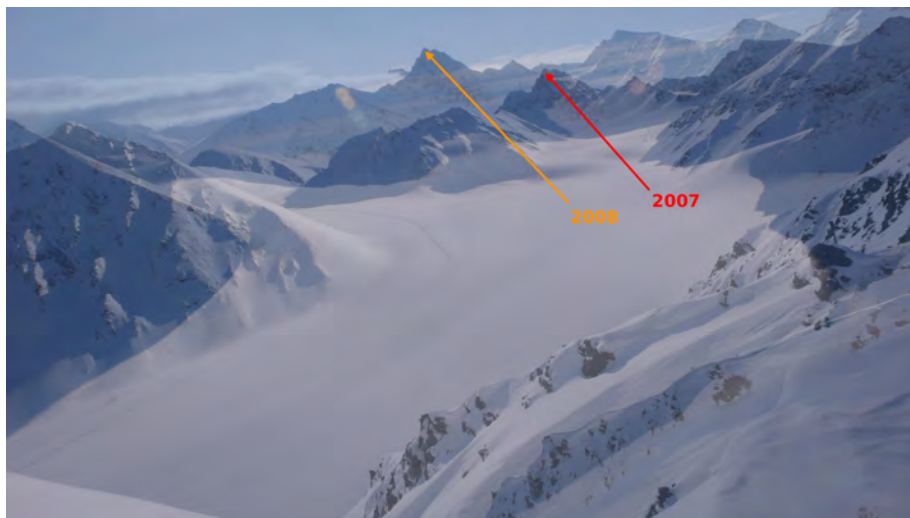


FIGURE 5.13 – Décalage du champ de vision d'une caméra entre 2007 et 2008

Nous avons remarqué que les variations, sauf dans le cas d'une caméra sujette aux rafales de vent, sont souvent par période (généralement entre deux maintenances). En plus des points de contrôle à définir pour le recadrage, il faut aussi détecter qu'il y a eu un changement de cadrage par rapport à l'image de référence. Dans notre cas, sur le même principe que la projection, nous avons défini un jeu de points de contrôle par caméra et par période. L'idéal aurait été de détecter les périodes de

temps et de définir les points de contrôle de façon automatique. Or, ceci est une problématique à part entière qui ne rentre pas dans le cadre de cette thèse. Donc nous nous sommes cantonnés à cette solution temporaire afin d'avoir un traitement automatique. Le résultat obtenu après le recadrage est une image de même dimension que l'image d'origine (figure 5.14). La zone perdue à cause du recadrage est modélisée en rouge étant donné qu'on ne peut pas reconstruire cette information.



FIGURE 5.14 – Image après recadrage. La zone rougeâtre est perdue.

### 5.3 Les limites des traitements

L'ensemble de ces traitements et prétraitements peuvent prendre du temps, et sont fastidieux s'ils sont fait à la main. Depuis la récupération des premières photos acquises par le programme HSF, certains traitements étaient semi-automatiques, d'autres totalement manuels, et certains étaient inexistantes. Les traitements n'étaient pas tous effectués avec le même logiciel, ce qui demandait un accès et une maîtrise de plusieurs logiciels. Toute la chaîne de traitement a été effectuée pour certaines dates particulières, notamment celles où le débit hydraulique n'était pas normal, mais la procédure de traitement mise en place à ce moment n'était pas du tout adaptée pour venir traiter plus de 60 000 images.

Une deuxième partie de la problématique de cette thèse est la définition de ces différents traitements afin de pouvoir traiter l'ensemble des photos *in situ* avec potentiellement des perturbations, tout en restant assez générique pour obtenir un outil capable de s'adapter aux traitements d'images *in situ* provenant de programmes similaires à celui de HSF.

## Chapitre 6

# La composition de services de traitements géographiques

The important thing is not to stop questioning. Curiosity has its own reason for existing.

---

*Albert Einstein*

Comme énoncé précédemment, l'un des avantages du *cloud* est d'offrir en tant que service des applications, des traitements et de la puissance de calcul et de stockage à travers Internet. Dans le cadre de l'analyse de l'évolution de la couverture neigeuse du glacier du Lovén Est au Spitsberg à partir de photos *in situ*, plusieurs traitements sont à mettre en place et à déployer en tant que service web afin de profiter des avantages du *cloud*. Les photos *in situ*, en fonction des perturbations subies, devront être traitées avec un nombre de services et un enchaînement de services variable.

### 6.1 Web Services

Afin de communiquer et de faire appel à un service via deux machines, éventuellement dotées de systèmes hétérogènes, Curbera [Curbera et al., 2001] définit deux aspects clés d'un service : un ensemble de standards minimum afin d'assurer l'interopérabilité élémentaire entre les machines et une représentation commune des services à travers des systèmes hétérogènes. De façon formelle, le *World Wide Web Consortium* (W3C) définit un *Web Service* (WS) comme suit :

DÉFINITION. — *A Web service is a software system identified by a Uniform Resource Identifier (URI), whose public interfaces and bindings are defined and described using Extensible Markup Language (XML). Its definition can be discovered by other software systems. These systems may then interact with the Web service in a manner prescribed by its definition, using XML based messages conveyed by Internet protocols. [Schlimmer, 2002]*

Comme le souligne la définition de W3C, un WS doit être "défini, décrit et découvert", clarifiant ainsi la notion "d'accessible" et rendant plus concret la notion de "orienté Internet" [Alonso et al.,



2003]. Non seulement les services doivent être opérationnels, mais encore ils doivent être décrits et annoncés de façon à rendre possible l'accès et l'interaction avec des clients. Bieber et Carpenter [Bieber and Carpenter, 2001] soulignent qu'un WS a un comportement et une utilisation régis par un contrat : il est vu "comme une boîte noire, l'implémentation étant cachée à l'utilisateur, la seule information qui lui est mise à disposition étant la description de l'action du service" [Le-grand Contes, 2011].

Dans l'objectif de construire une *Service-Oriented Architecture* (SOA), les WS s'appuient sur quatre technologies (résumées en figure 6.1) :

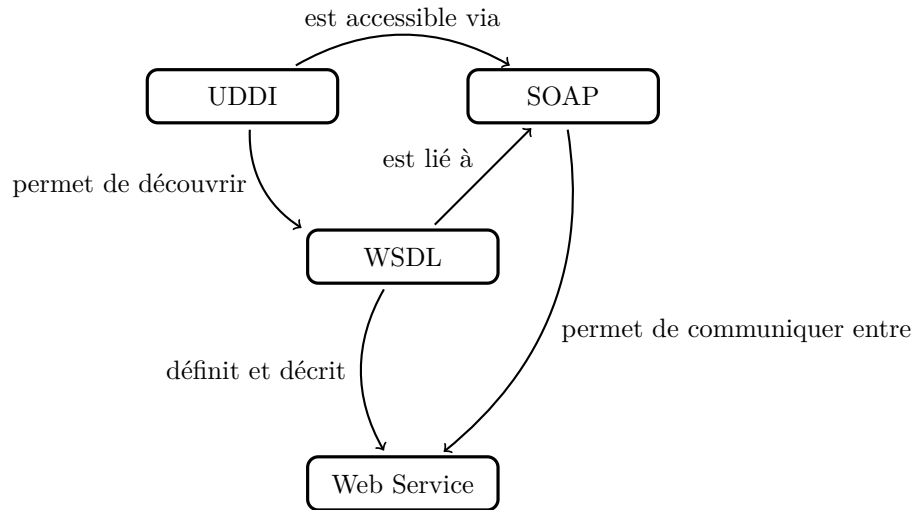


FIGURE 6.1 – Relation entre les technologies utilisées pour les WS. Adaptation de [Erl, 2008]

1. **La description des données : XML Schema Definition (XSD)** est un méta-langage XML utilisé pour définir la structure d'un document XML. Dans le contexte des WS, il définit la structure des données échangées entre services.
2. **La description des fonctionnalités : Web Service Description Language (WSDL)**, implémenté en XML, décrit les services comme un ensemble de *endpoints*<sup>1</sup> opérant avec des messages contenant soit des informations orientées document ou orientées procédure (traitement) [Christensen et al., 2001]. Le WSDL permet de définir les fonctionnalités du service en terme d'opérations disponibles, de spécifier le protocole de communication ainsi que le format de messages requis pour communiquer, et d'indiquer la localisation du service.
3. **La communication : Simple Object Access Protocol (SOAP)** est un protocole léger destiné à l'échange d'informations structurées, appelées messages, dans un environnement décentralisé et distribué [Gudgin et al., 2007], notamment utilisé pour les WS. Ces messages, envoyés sous forme de documents XML, sont transportés sur le réseau en utilisant des protocoles standards. Le protocole de transport le plus utilisé est *Hypertext Transfer Protocol* (HTTP) qui a pour avantage d'être utilisé largement sur Internet et d'être indépendant de

1. Un *endpoint* est l'*Uniform Resource Locator* (URL) par laquelle un client peut accéder au service.

la plateforme. Les messages SOAP sont composés d'une enveloppe, qui définit la structure du message et la façon de le traiter, d'une entête, qui correspond à un ensemble de règles de codage pour exprimer les types de données définis par l'application, et d'un corps, qui contient l'information nécessaire à l'appel du service (l'opération à invoquer et les paramètres à lui fournir).

4. **La publication : *Universal Description Discovery and Integration* (UDDI)** est un standard qui assure la publication et la découverte de services dans une SOA [Bellwood et al., 2004]. Il correspond à un annuaire de services qui publie les interfaces des WS destinées aux consommateurs de services, en se basant sur les descriptions de services spécifiées en WSDL. Cet annuaire est divisé en trois catégories [Hassen, 2009] : les pages blanches qui contiennent les informations sur les fournisseurs de services (le nom de l'entreprise, ses coordonnées), les pages jaunes qui cataloguent les WS de chacune des entreprises sous le format WSDL, et les pages vertes qui fournissent des informations techniques précises sur les services fournis.

La capacité des WS à se regrouper (plusieurs WS développés par différentes organisations) afin de répondre à une exigence d'un client les rend attractifs. Un tel regroupement est basé sur la norme commune des interfaces des WS, sans prendre en considération les langages dans lesquels ils ont été implémentés ou les plateformes où les services seront exécutés [Rao, 2004]. Afin qu'une telle programmation basée sur les services soit utilisable, il faut respecter un ensemble de principes par rapport aux services [Erl, 2008] :

**Un langage standard pour la description d'un service** : Comme nous avons pu le voir jusqu'à présent, la description du service est la partie visible par tous les clients et elle permet de décrire les fonctionnalités proposées ainsi que la manière d'y accéder. Même si les services sont implémentés dans différents langages de programmation et sont exécutés sur différentes plateformes, la description permet de définir une interface standardisée des fonctionnalités, et permet ainsi la création de messages standardisés.

**Le faible couplage** : Dans le développement d'applications, le couplage correspond au degré de dépendance qui existe entre les différents composants/modules de l'application. Avoir un degré de couplage faible est une notion clé de la démarche de services. Ceci leur permet d'être autonomes et de fonctionner indépendamment les uns des autres, leur assurant des changements (localisation, implémentation, fournisseur) sans engendrer de problèmes sur les autres entités de l'architecture. Cela permet également aux WS de se trouver et de communiquer entre eux dynamiquement lors de l'exécution.

**Abstraction, autonomie** : L'abstraction souligne l'importance de cacher le plus possible les détails d'un service. Ceci permet d'établir et de préserver le principe de faible couplage. Un service est cette boîte noire autonome qui ne contrôle que la logique métier qu'il encapsule.

**Ré-Utilisabilité, composabilité** : Un service correspond à une brique logiciel réutilisable. Il est réalisé en partageant la logique de l'application afin qu'il puisse être potentiellement utilisé par plusieurs clients. Cette propriété de réutilisabilité nous permet d'accéder à celle de composabilité. En effet, étant réutilisable, les services peuvent être regroupés dans des services composites qui coordonnent des échanges de données entre-eux.

**Sans état** : Afin de pouvoir conserver les principes de réutilisabilité et de composabilité, il est conseillé que les services respectent la propriété d'idempotence : l'exécution d'un service ne doit pas dépendre d'un état antérieur et les services ne conservent pas un état spécifique après

une exécution. Ceci n'empêche pas pour autant que le service interagisse avec des données persistantes et que ces dernières influencent son exécution.

## 6.2 La composition de services

Pendant ces dernières années, l'un des défis de la SOA est l'intégration de services pour l'approvisionnement de nouveaux services personnalisés et enrichis. Si une application ou un client requiert des fonctionnalités et qu'aucun service ne peut répondre directement à ce besoin, il devrait être possible de combiner ou de composer des services existants afin de répondre aux besoins de cette application ou ce client [Matskin et al., 2005, Preist et al., 2001, Benatallah et al., 2005, Alonso et al., 2003].

**DÉFINITION.** — *La composition de services vise à faire interopérer, interagir et coordonner plusieurs services pour la réalisation d'un but. [Charif, 2007]*

La composition de services est un processus qui permet de passer d'une spécification abstraite de la composition vers une description exécutable. Le service composite joue le rôle de *coordonnateur de services* en définissant comment les services doivent être combinés, synchronisés et coordonnés. Yang et Papazoglou [Yang and Papazoglou, 2004] ont défini quatre phases pour la composition de services :

1. **Préparation, définition abstraite** : permet de déterminer la série de fonctionnalités des services qui seront nécessaires, de les récupérer puis de les regrouper afin de générer le service demandé par le client. Lors de la préparation, les différentes fonctionnalités sont identifiées et les interactions sont spécifiées.
2. **Planification** : sert à déterminer comment et quand les services seront exécutés. Lors de la planification, la conformité et la compatibilité des services sont vérifiées.
3. **Construction** : fournit une composition concrète prête à s'exécuter. Seulement les services potentiellement disponibles lors de l'exécution sont à déterminer.
4. **Exécution** : implémente les liaisons avec les différents services disponibles puis exécute la composition.

La liaison avec les services peut être faite à différents endroits de la conception. On parlera de composition *statique*, *semi-statique* (ou *semi-dynamique*) ou *dynamique*.

**Composition statique** : les services constituant la composition sont pré-définis et les liens vers les fonctionnalités des services sont sélectionnés avant l'exécution.

**Composition semi-statique ou semi-dynamique** : contrairement à la composition statique, le lien vers les fonctionnalités se fait à l'exécution. Mais une fois effectuées, les liaisons restent généralement figées.

**Composition dynamique** : les services constituant la composition sont repérés et intégrés à la volée, en fonction de leur apparition ou disparition pendant l'exécution.

Le modèle de composition dans le temps est généralement modélisé par un graphe, où chaque sommet correspond à l'invocation d'une fonctionnalité d'un service, et les arcs servent à représenter l'ordre d'enchaînement des opérations. La composition est décrite dans un langage qui est interprété par un moteur d'exécution. La responsabilité du moteur est d'invoquer les services dans l'ordre

spécifié, de réaliser le routage des données, de maintenir et de gérer l'état des activités et du composite, ainsi que de gérer les situations d'exception.

Deux approches existent pour ce type de composition (illustration en figure 6.2) [Peltz, 2003, Legrand Contes, 2011, Charif, 2007] :

### L'orchestration de services

DÉFINITION. — *L'orchestration décrit, du point de vue d'un service, les interactions de celui-ci ainsi que les étapes internes (ex. transformations de données, invocations à des modules internes) entre ses interactions. [Peltz, 2003]*

L'orchestration décrit l'interaction des services impliqués dans la composition, les messages qu'ils échangent, l'ordre et la manière par laquelle les services interagissent entre eux (par exemple, en séquence, en parallèle, sous conditions). Le service composite, appelé également le chef d'orchestre, permet d'enchaîner des services d'une manière prédéfinie. Il agit comme une entité centralisée qui coordonne les interactions entre services.

### La chorégraphie de services

DÉFINITION. — *La chorégraphie décrit la collaboration entre une collection de services dont le but est d'atteindre un objectif donné. L'accomplissement de ce but commun se fait alors par des échanges ordonnés de messages. [Austin et al., 2004]*

La chorégraphie décrit d'un point de vue global la façon dont un groupe de services collabore. Elle décrit les interactions auxquelles les services sollicités participent afin d'atteindre cet objectif, ainsi que les dépendances entre les interactions telles que le flot de contrôle (par exemple, si une interaction doit en précéder une autre), le flot de données, les correspondances des messages, les contraintes de temps, les dépendances transactionnelles, etc. Contrairement à l'orchestration, la chorégraphie ne centralise pas le contrôle de l'invocation des services mais elle présente plutôt une vision globale de tous les participants relativement à la manière dont ils doivent participer à la réalisation d'un but commun.

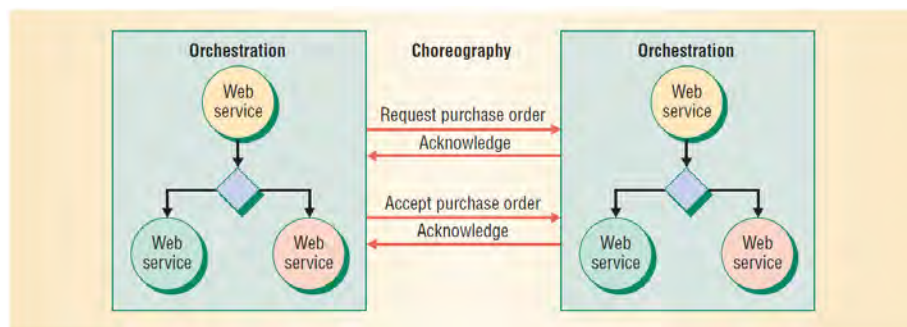


FIGURE 6.2 – Orchestration vs chorégraphie. L'orchestration se réfère à un processus exécutable. La chorégraphie suit les séquences de messages entre les groupes et les sources. Source : Peltz [Peltz, 2003]

## 6.3 Méthodes de composition de services

Mettre en place un service composé d'autres services de manière automatisée ou semi-automatisée est la clé du succès des SOA. Afin d'y parvenir, plusieurs méthodes ont été proposées.

La définition d'un service composite comprend un ensemble de services et le flux de données entre ces services. De façon similaire, un *workflow* spécifie l'enchaînement des différentes tâches. L'accomplissement des travaux sur la flexibilité des *workflows*, l'adaptation automatique des processus et l'intégration inter-entreprise procure également des possibilités pour l'automatisation de la composition de WS. De plus, les méthodes dynamiques de *workflows* permettent de faire le lien entre les noeuds abstraits et les ressources concrètes, ou les services, de façon automatique [Lin, 2012, Jaradat et al., 2013, van der Aalst and ter Hofstede, 2005].

D'un autre côté, les méthodes dynamiques de composition sont requises pour planifier automatiquement. La plupart des méthodes de cette catégorie sont apparentées aux méthodes de planification de l'intelligence artificielle et de la déduction de théorèmes. Une hypothèse générale par rapport à ce type de méthodes est que chaque WS peut être spécifié par ses pré-conditions et ses post-conditions dans le contexte de planification.

Dans ce qui suit, nous allons présenter différentes méthodes existantes pour mettre en place un service composite. Ces méthodes peuvent être soit statiques, semi-dynamiques ou totalement dynamiques.

### 6.3.1 Composition de services via le business process

Le succès des WS est principalement dû au développement des standards par des organismes tel que le W3C. Ces organismes continuent de développer et mettre en place des standards de certaines notions clés des WS comme la description des interfaces, la sécurité et les pratiques de gestion de services et à plus haute échelle, pour l'implémentation de *business processes* [Srinivasan and Treadwell, 2005].

Le *Business Process Execution Language* (BPEL) est un langage qui permet de spécifier le comportement d'un *business process* basé sur des WS. BPEL est, de nos jours, le standard pour la modélisation des *business processes* en entreprises. Dans le domaine de la recherche scientifique, aucune préférence n'est clairement définie, mais de nombreux chercheurs commencent à utiliser BPEL comme langage de composition [Dornemann et al., 2009, Karande et al., 2011].

Le BPEL est basé sur une structure XML, dont certaines balises sont obligatoires pour son bon fonctionnement (Listing 6.1).

Listing 6.1 – "Structure d'un BPEL"

```
<process name="BPName" targetNamespace="anyURL"
  xmlns="http://schemas.xmlsoap.org/ws/2003/03/business-process/">
  <partnerLinks>
    <!-- The declaration of partner links -->
  </partnerLinks>

  <variables>
    <!-- The declaration of variables -->
  </variables>
```

```
<sequence>
  <!-- The definition of the BPEL business process main body -->
</sequence>
</process>
```

**process** : cet élément permet d'assigner un nom et un espace de nom associé au service composite qui sera créé à travers ce BPEL.

**partnerLinks** : cet élément est composé d'autant de balises *<partnerLink>* que de services auxquels il fera appel pendant l'exécution du *business process*.

**variables** : cet élément est composé d'autant de balises *<variable>* que de variables nécessaires à l'échange de messages entre les différents services. Chaque service appelé aura besoin de deux messages : celui de l'appel et celui de la réponse. Les variables permettant cet échange ont en paramètre le type de donné qui sera transmis ainsi que la valeur de la donnée si besoin.

**sequence** : chaque *business process* décrit dans son corps principal la manière dont chacune des fonctionnalités des services sera appelée. Il commence généralement par la balise *<sequence>* pour définir un flux séquentiel d'appel, mais ce peut être également *flow* ou *switch* si les appels aux services sont effectués en parallèle ou en fonction de différents cas définis. Le corps principal est composé d'appels à des services, spécifiés par les balises *<invoke>* et *<receive>*, ainsi que d'affectations des variables nécessaires à l'appel aux différents services par la balise *<assign>*.

BPEL met à disposition encore d'autres fonctionnalités pour la mise en place d'un *business process*, telles que les boucles, la gestion des exceptions ou la mise en place de mécanisme pour faire un *roll-back*.

Par définition, BPEL correspond à un modèle de composition statique étant donné que l'enchaînement des services est déjà connu et fixé avant l'exécution. D'autres méthodes, basées sur la structure du BPEL tout en étant couplées à des méthodes de logiques telles que les réseaux de petri ou le *pi-calcul*, ont permis de rendre plus dynamique la composition de services [Lohmann, 2007, Lucchi and Mazzara, 2007].

### 6.3.2 Composition de services via la coordination d'agents

Dans le contexte de la composition de services, ajouter de la flexibilité réside dans la capacité de choisir dynamiquement les services pendant l'exécution plutôt que de se baser sur un choix fixe d'un *workflow*.

Les agents intelligents et les technologies multi-agents permettent de varier les approches qui peuvent être réalisées afin d'améliorer les solutions existantes de la composition de services à travers différents systèmes de négociations et de commissaires [Preist et al., 2003, Sycara et al., 2004, Müller et al., 2006].

DÉFINITION. — *Un agent communique avec ses pairs en échangeant des messages dans un langage de communication entre agents explicite. Même les agents peuvent être aussi simples qu'une sous-routine, ce sont typiquement des entités plus larges avec une sorte de contrôle permanent. [Genesereth and Ketchpel, 1994]*

De façon plus générale, la définition d'un agent peut se faire à travers ses différentes propriétés système [Wooldridge and Jennings, 1995, Charif, 2007, Ferber and Perrot, 1995] :

**autonome** : les agents opèrent sans intervention directe d'un humain ou autre, et ont une sorte de contrôle sur leurs propres actions et leur propre état [Castelfranchi, 1994] ;

**sociable** : les agents interagissent avec d'autres agents (potentiellement des humains) à travers un langage de communication inter-agent [Genesereth and Ketchpel, 1994] ;

**réactif** : les agents ont une perception de leur environnement et répondent de façon opportune aux changements qui y apparaissent ;

**pro-actif** : les agents n'agissent pas simplement à une réponse face à un changement de leur environnement, ils sont également capables de montrer un comportement orienté en prenant des initiatives.

Partageant le même environnement, les agents sont amenés à collaborer pour accomplir leurs objectifs en s'entraïdant, à partager leurs ressources, à résoudre ou à éviter des conflits éventuels, etc. Ceci amène les agents à se coordonner pour évoluer en complète synergie dans cet environnement. Par conséquent, afin de déterminer l'organisation la plus appropriée et communiquer les résultats de leur coopération, les agents interagissent entre-eux à travers différents mécanismes de négociations.



FIGURE 6.3 – Structure d'un agent offrant un service. Source : Charif [Charif, 2007]

La composition de services et les systèmes multi-agents partagent plusieurs propriétés telles que la complexité du problème global à résoudre, la flexibilité des interactions et un environnement dynamique. En greffant aux WS les fonctionnalités d'un agent (figure 6.3), la chorégraphie dynamique de services semble être réalisable par des systèmes multi-agents [Huhns, 2002, Jonquet, 2006, Bourdon, 2007].

### 6.3.3 Composition de services via la planification

Plusieurs recherches se sont également dirigées vers les problèmes de planification pour la composition de services. De manière générale, un problème de planification peut être représenté à travers un quintuplet  $\langle S, S_0, G, A, \Gamma \rangle$ , où [Rao et al., 2004] :

- $S$  représente l'ensemble des états possibles de l'univers de la composition de services,
- $S_0 \subset S$  correspond à l'état initial,

- $G \subset S$  correspond à l'état final que le système de planification tente d'atteindre,
- $A$  est l'ensemble des actions possibles que le planificateur peut appeler dans l'objectif de changer l'état courant vers un autre état de  $S$ ,
- $\Gamma \subseteq S \times A \times S$  est la relation de translation qui détermine les pré-conditions et les post-conditions relatives à l'exécution de l'action.

Dans le cadre de la planification pour la composition de services,  $A$  correspond à l'ensemble des services accessibles,  $S$  à l'ensemble des états nécessaires et générés par les services et  $\Gamma$  à l'ensemble des changements d'états suite à l'appel d'un service. Ces différentes informations concernant le type de changements effectués par une fonctionnalité d'un service ou les différents états ont besoin d'être renseignés pour la planification.

La sémantique, et tout particulièrement le Web sémantique [Berners-Lee et al., 2001] qui est une extension du Web, permet de rajouter du sens aux différentes informations présentes, facilitant ainsi la coopération dans les tâches entre ordinateur et client. Même si l'utilisation de la sémantique est très utilisée par les méthodes de planification, elle peut également être introduite dans les méthodes de business process ou de coordination d'agents afin d'ajouter du sens aux données et d'utiliser des méthodes issues de la logique et des graphes pour trouver de nouveaux moyens pour collaborer entre services. L'ajout de la sémantique sur les informations telles que les propriétés et les relations se fait par annotation. Un ensemble d'outils a été développé à cet effet [Lecué, 2008].

**Resource Description Framework (RDF)** : est un langage d'assertion, généralement implémenté en XML, qui permet de représenter l'information sous forme de triplet [Brickley and Guha, 2014]. Ces triplets sont formés d'un sujet, représenté par une URI, d'un prédicat et d'un objet, représenté soit par une URI soit par un littéral (chaîne ou nombre). L'ensemble de ces triplets correspond à un *RDF Schema* (RDF-S), qui procure de l'information par rapport aux interprétations des affirmations données à travers un modèle de données en RDF.

**Web Ontology Language (OWL) (anciennement DAML)** est un langage de web sémantique standardisé qui étend les possibilités du RDF [Antoniou and Van Harmelen, 2004, McGuinness et al., 2012]. Il implémente toute la logique descriptive et exprime des relations plus complexes que le RDF telles que l'équivalence, l'existence, les combinaisons logiques (union, intersection ou complémentarité), la restriction ou la symétrie.

Afin de pouvoir manipuler automatiquement les services, ces derniers sont annotés sémantiquement à travers le *Semantic Annotation for WSDL* (SAWSDL) [Farrell and Lausen, 2007, Kopecný et al., 2007] dans l'objectif de leur donner plus de sens. Les descriptions des services annotés contiennent des balises supplémentaires dans le WSDL afin de les référencer à un modèle sémantique tel que les triplets RDF.

Plusieurs méthodes de composition de services via la planification existent, et nous allons uniquement parler des plus fréquentes ces dernières années [Rao et al., 2004, Lecué, 2008].

- **Calcul des situations** : McIlraith et son équipe [McIlraith et al., 2001, McIlraith and Son, 2002, Narayanan and McIlraith, 2002] ont travaillé sur l'adaptation et l'extension du langage Golog pour la composition automatique de services. Golog est un langage de programmation logique basé sur le calcul de situation. Par la mise en place de procédures génériques de haut niveau et de contraintes personnalisées, les auteurs répondent à la problématique de composition de services.



- **Planning Domain Definition Language (PDDL)** : Rendu connu par McDermott [McDermott et al., 1998], il est devenu une norme pour la représentation et l'échange de modèles de planification. Le langage PDDL fournit une base sur laquelle une norme peut être construite, permettant à travers un modèle de partager et de motiver le développement du domaine de la planification vers l'application réaliste [McDermott, 2002, Klusch and Gerber, 2005]. En présentant la méthode de composition de services basée sur le PDDL, McDermott introduit un nouveau type de connaissance, appelée valeur d'une action, qui persiste et qui n'est pas traitée comme une vérité littérale. Du point de vue de la construction de WS, la fonctionnalité nous permet de distinguer la transformation de l'information et le changement d'état produit par l'exécution du service.
- **Planification basée sur des règles** : Il s'agit d'une technique pour générer des services composites à partir d'une description de haut niveau [Medjahed et al., 2003, Beauche and Poizat, 2008]. Cette méthode utilise des règles pour déterminer si deux services sont composables. L'approche de la composition est constituée de quatre phases. Premièrement, la phase de spécification, qui permet une description de haut niveau des compositions désirées. Deuxièmement, la phase de correspondance, qui utilise des règles de composition pour générer des plans de composition qui sont conformes à la spécification décrite par le client. Troisièmement, la phase de sélection : si plus d'un plan est généré, le client sélectionne un plan basé sur la qualité de la composition (QoC) telle que le rang, le coût, etc. Quatrièmement, la phase de génération : une description détaillée du service composite est générée automatiquement et présentée au client.
- **Autres méthodes** : Plusieurs autres techniques d'intelligence artificielle sont proposées pour la composition de services. Nous n'en énumérerons qu'une partie.

Depuis le début des années 1980, la planification en intelligence artificielle s'est basée sur la décomposition en *Hierarchical Task Network* (HTN)[Erol et al., 1994]. Vingt ans plus tard, un nouveau système de planification basé sur HTN a été remarqué notamment pour ses performances : *Simple Hierarchical Ordered Planner* (SHOP) [Nau et al., 1999] puis sa nouvelle version SHOP2 [Sirin et al., 2004, Nau et al., 2003]. SHOP2 génère les étapes de chaque plan dans le même ordre que celles-ci seront ensuite exécutées. Il connaît donc l'état courant à chaque étape du processus de planification. D'après les travaux de Wu [Wu et al., 2003], la planification obtenue par SHOP2 est utilisable pour la composition de services.

Lämmermann [Lämmermann, 2002] a mis au point une méthode de composition qui est adaptée aux services. La méthode intègre une technique efficace de synthèse déductive avec un langage de spécification. Ce travail est réalisé dans le cadre d'une plate-forme logicielle orientée objet. La plate-forme présente une extension déclarative à un langage de programmation orienté objet, afin de combiner les avantages de la programmation orientée objet avec les côtés positifs de la programmation déclarative.

Lajmi [Lajmi et al., 2007], dans le cadre de la problématique de la composition de services web dans un contexte de systèmes ouverts et dynamiques, a pour objectif de trouver une solution intelligente pour la composition de WS. Afin de rendre la composition semi-automatique, il propose l'utilisation de raisonnement à partir de cas comme une technique d'apprentissage.

## 6.4 La composition pour la géographie

Comme nous avons pu le voir dans le chapitre précédent, l'analyse de l'évolution de la couverture neigeuse génère un grand nombre de données qui doivent être traitées. Les traitements sont complexes et avant d'aboutir au résultat final, les photos *in situ* nécessitent un nombre variable de traitements.

En implémentant l'ensemble des traitements cités précédemment en tant que WS, ceci nous donne plusieurs opportunités :

- En tant que WS, les traitements pourront être modulables et évolutifs à tout moment, sans pour autant perturber l'ensemble de la chaîne de traitements ;
- une première composition de services, correspondant à l'enchaînement actuel des traitements, peut correspondre à un premier résultat ;
- les différentes notions de composition dynamique ou semi-dynamique peuvent venir améliorer le premier résultat ;
- les différentes perturbations du terrain capturées par les photos *in situ* peuvent être traitées individuellement par une composition personnalisée.

Les premières études menées sur les photos *in situ* lors du programme HSF nous apportent une connaissance sur le processus d'enchaînement des traitements non négligeable. Afin d'en tirer parti, nous avons choisi une composition formalisée en BPEL. Cependant, une composition statique ne convient pas à la problématique d'adaptation des traitements en fonction des perturbations subies. Nous avons donc choisi d'enrichir les services par des annotations sémantiques et, grâce à une ontologie<sup>2</sup> sur les services, de générer de nouvelles compositions afin de s'adapter à l'état des photos. Nous détaillerons ceci dans le chapitre 9.

---

2. Une ontologie est un réseau sémantique qui regroupe un ensemble de concepts décrivant complètement un domaine. Ces concepts sont liés les uns aux autres par des relations taxinomiques (hiérarchisation des concepts) d'une part, et sémantiques d'autre part.



## Chapitre 7

# La géographie, laboratoire d'essai à grandeur nature pour le *cloud*

Once we accept our limits, we go beyond them.

---

*Albert Einstein*

Depuis la création de la première machine, l'informatique a beaucoup évolué, et tout particulièrement depuis ces trois dernières décennies. À partir des années 90, l'ordinateur s'est infiltré dans tous les milieux et dans tous les systèmes (technique, financier, administratif, etc.). Avec la numérisation des différents supports (photos, son, documents, etc.), et l'évolution des composants de l'ordinateur (processeur, mémoire, etc.), l'informatique devient omniprésente dans l'activité humaine. Couplé à l'évolution des composants et des supports, le développement de nouveaux logiciels, adaptés aux besoins de chaque utilisateur dans chacun de leurs domaines respectifs, explose. Plusieurs recherches sont effectuées sur les différents aspects liés à l'informatique : l'interaction homme-machine, la structure de l'implémentation d'un logiciel en fonction du type de langage, les architectures parallèles et distribuées, le réseau, les algorithmes et les théories de nombreux domaines (graphes, images, logique, etc.), etc [Hassenzahl et al., 2000, Jähne et al., 1999, Kon et al., 2002, Tanenbaum and Van Steen, 2007, Wolfgang, 1994, Jacobson et al., 1999, Kumar et al., 1994, Ahuja et al., 1993, Wolkowicz et al., 2012, Gross and Yellen, 2004, Glover and Kochenberger, 2006, Baader, 2003].

Cependant, la recherche est parfois limitée à certaines notions uniquement théoriques à cause d'un manque de données concrètes (en grande quantité et entachées d'imperfections) qui permettent tester les méthodes dans des cas réels. La collaboration avec des géographes nous donne l'occasion de tester notre modèle avec des données réelles.

Pour le moment, notre étude ne s'est intéressée qu'au traitement des photos *in situ* pour l'analyse de l'évolution du manteau neigeux. Comme nous avons pu le voir dans le chapitre 5, le processus de traitement des photos est complexe. Jusqu'à présent, certains traitements étaient semi-automatiques, d'autres totalement manuels et certains étaient inexistantes. Le travail de cette thèse s'est concentré sur **la mise au point d'un modèle d'application qui permettrait de centraliser l'appel au traitement des photos à l'aide d'un seul outil le plus automatique**

**possible.** Puisque certains traitements sont inexistantes et dans l'objectif de mettre en place un outil facilement évolutif, il était important que le modèle d'application proposé soit dynamique. Ainsi, lorsqu'un nouvel algorithme est créé afin de faire un traitement particulier, celui-ci peut être intégré au modèle après son implémentation en tant que service web. De la même façon, si l'analyse du paysage nécessite un autre type de traitement de données afin d'en extraire un autre type d'information (par exemple, glissement du manteau neigeux sur la glace), il pourrait également être intégré à l'outil à condition d'implémenter les traitements associés à cette nouvelle fonctionnalité souhaitée. De plus, même si nous nous sommes focalisés sur le traitement de photos de glacier, le même processus de collecte d'information à partir de stations photo automatiques pourrait être mis en place sur d'autres sites et pour l'analyse d'autres paysages (par exemple, l'évolution du désert du Sahara, ou l'érosion des sols d'Asie à cause d'une culture intensive de riz). Même si le paysage est différent et que les éléments composants l'analyse ne sont pas de la neige et de la glace, les traitements à appliquer restent les mêmes (modulo les perturbations spécifiques du lieu d'acquisition) : projection, classification, mosaïque. Il est donc important que l'implémentation des traitements soit générique afin de pouvoir intégrer tous les cas de figure similaires. Dans le chapitre 8, nous définissons le modèle d'application, basé sur les WS et le *cloud computing*, que nous avons mis en place afin de traiter toutes les données automatiquement. Afin de pouvoir s'adapter aux différents changements et aux différentes perturbations, nous avons détaillé dans le chapitre 9 l'ordonnanceur de services que nous avons mis en place. À travers ces deux chapitres, nous définissons notre approche et notre solution technologique face à nos objectifs.

Les données récupérées par le projet HSF nous ont permis de tester notre architecture pour le traitement des photos ainsi que pour des méthodes de réduction de dimension des données. En effet, cette quantité de données est à la fois une mine d'or pour l'analyse environnementale, mais elle est une difficulté pour le traitement tout comme pour le stockage. Dans les deux derniers chapitres, nous présentons nos expériences pour le traitement comme pour le stockage ainsi que les résultats obtenus.

Dans le chapitre 10, nous comparons notre modèle de traitement de données sur trois architectures différentes et avec un nombre d'images variable afin de voir comment les différentes architectures arrivent à supporter la charge et mettre en évidence l'intérêt du *cloud* pour le traitement d'une grande quantité de données.

Dans le chapitre 11, nous présentons et comparons les méthodes de réduction de dimension basées sur les composantes principales et les composantes indépendantes.

Deuxième partie

L'architecture et le modèle



The grand aim of all science is to cover  
the greatest number of empirical facts by  
logical deduction from the smallest  
number of hypotheses or axioms.

---

*Albert Einstein*





## Chapitre 8

# Modèle d'application dans le *cloud*

We cannot solve our problems with the same thinking we used when we created them.

---

*Albert Einstein*

### 8.1 Introduction partielle

Analyser l'évolution de la couverture neigeuse d'un glacier nécessite un suivi régulier du glacier. L'utilisation d'images satellites pour le suivi du glacier du Lovén Est n'a pas été suffisante car, entre le coût important de l'acquisition d'un cliché et le peu d'images utilisables à cause des mauvaises conditions météorologiques à ces latitudes, la fréquence des clichés est d'environ un mois.

Étant donné que l'évolution de la couverture neigeuse peut être constituée d'événements rapides tels que la fonte des neiges ou l'enneigement, la fréquence d'un mois est trop grande pour détecter ces évolutions. Nous avons donc pallié au manque de clichés satellites par un ensemble de clichés *in situ*. Tout autour du glacier du Lovén Est, six appareils ont été mis en place avec un micro-contrôleur afin de les rendre autonomes. Ces appareils ont été programmés pour prendre trois clichés par jour : 8h, 12h et 16h.

À l'opposé des images satellites, les photos *in situ* ne capturent qu'une partie du glacier. Leur fréquence d'acquisition nous permet d'avoir un suivi beaucoup plus régulier en générant beaucoup de données, mais, à cause de l'autonomie des appareils, les photos sont parfois perturbées par les conditions météorologiques.

Afin de cartographier et quantifier l'évolution de la couverture neigeuse au cours du temps, nous reconstituons un équivalent de l'image satellite à partir des différentes photos *in situ*. L'ensemble de ces traitements sur les photos correspond à la projection des images sur un plan, à la construction d'une mosaïque afin de reconstituer l'ensemble du glacier, puis classifier les différents états du glacier : neige, glace, roche. À cause des différentes perturbations subies par les appareils autonomes, un certain nombre de pré-traitements peuvent être à appliquer également (plus de détails dans le chapitre 5).

Dans cet article, nous présentons un modèle basé sur un environnement de *cloud computing*, avec chacun des traitements définis précédemment comme étant des WS. La cartographie de l'état du glacier (neige, glace, roche) correspond à l'enchaînement de ces différents services sur chacune des photos.

Ce modèle permet d'obtenir un outil de traitement de données qui est générique et dynamique, et qui permet de répondre aux problématiques de variétés et quantités de photos à devoir traiter de façon automatique.

## **8.2 Article "*A dynamic and generic cloud computing model for environmental analysis using in situ data applied to glacier mass balance*"**



Contents lists available at ScienceDirect

# International Journal of Applied Earth Observation and Geoinformation

journal homepage: [www.elsevier.com/locate/jag](http://www.elsevier.com/locate/jag)

## A dynamic and generic cloud computing model for glaciological image processing



Élisabeth Ranisavljević<sup>a,\*</sup>, Florent Devin<sup>a</sup>, Dominique Laffly<sup>b</sup>, Yannick Le Nir<sup>a</sup>

<sup>a</sup> *École Internationale des Sciences du Traitement de l'Information, 2 boulevard Lucien Favre, CS 77563, 64 075 Pau Cedex, France<sup>1</sup>*

<sup>b</sup> *University of Toulouse, GEODE UMR 5602 CNRS, Maison de la recherche, 5 allées Antonio Machado, 31 058 Toulouse Cedex, France*

### ARTICLE INFO

#### Article history:

Received 20 December 2012

Accepted 9 August 2013

#### Keywords:

Environment

*in situ* sensing

Image

Cloud computing

Scheduler

Web service

### ABSTRACT

As satellite imaging is quite expensive, and because of poor weather conditions including common heavy cloud cover at polar latitudes, daily satellite imaging is not always accessible or suitable to observe fast temporal evolutions. We complement satellite imagery with a set of ground based autonomous automated digital cameras which take three pictures a day. With these pictures we build a mosaic with their projection and apply a classification to define the temporal evolution of the snow cover. As the pictures are subject to heavy disturbance, some processing is needed to build the mosaic. Once the processes are defined, we present our model. This model is built upon a cloud computing environment using Web services workflow. Then we present how the processes are dynamically organized using a scheduler. This scheduler chooses the order and the processes to apply to every picture to build the mosaic. Once we obtain a mosaic we can study the variation of the snow cover.

© 2013 Elsevier B.V. All rights reserved.

### 1. Introduction

Polar regions and especially glaciers are considered as relevant indicators of climate change (Bernard, 2011). In 2007, the International Polar Year coordinated many different projects focused on the Arctic and Antarctic dynamics. One of these projects was the Hydrow-Sensor-FLOW (HSF), supervised by M. Griselin, C. Marlin and D. Laffly (Laffly et al., 2012). Its study area is located on the west coast of Spitsberg (Svalbard) and focused on the East Lovén glacier (Fig. 1). HSF aims to follow the spatiotemporal dynamics of the East Lovén glacier to understand its reactivity to climate change. HSF combines the disciplines of hydrology, glaciology, climatology and snow science. For this project, a rich database has been designed with temperature, precipitation, remote sensing imagery, *in situ* pictures, digital model elevation, snow density, etc. Environmental analysis is obtained by linking all the analyses from each expert together: water balance, classification of the different glacier states, etc. (Bernard, 2011).

One of the goals of the HSF is to map the temporal evolution of the snow cover of a glacier. In order to process it, we need to collect images of the whole glacier and then classify the different states (snow and ice). Due to poor weather conditions including

common heavy cloud cover at polar latitudes and because of its cost, daily remote sensing images are not always accessible. Moreover, fast events like floods or blankets of snow are missed by satellite based studies, since the slowest sampling rate is unable to observe them. In order to overcome this problem, a set of six ground based autonomous<sup>2</sup> digital cameras has been installed in the glacier, covering the whole area (Corripio, 2004; Hinkler et al., 2002; Laffly et al., 2012; Newbery and Southwell, 2009). The cameras are commercial off-the-shelf (either Leica D-Lux4 or Panasonic Lumix LX3), selected for their optical lens, high sensor resolution (10Mpixel sensors) and data storage on Secure Digital mass storage memory card. They have been customized (e.g., airtight enclosure and microcontroller) in order to support the hard polar climate conditions. The ground datasets provide a high spacial resolution of 10–100 cm/pixel side (Laffly et al., 2012). An example of such a concept is given in Fig. 2. As the cameras are at low altitude, all under 550 m, they are regularly below the clouds. The cameras have been programmed to take three pictures a day (08:00, 12:00 and 16:00) since summer 2007 to complement satellite imagery. With the different *in situ* pictures, a map projection can be built up by mosaicking and georeferencing data.

The actual database contains 13 415 images (65 Gb). These pictures form a very large database, but there is noise because of weather conditions, electronic deficiency and geometry variations.

\* Corresponding author. Tel.: +33 559059091.

E-mail addresses: [erc@eisti.eu](mailto:erc@eisti.eu), [elisabeth.ranisavljevic@eisti.eu](mailto:elisabeth.ranisavljevic@eisti.eu) (É. Ranisavljević), [fd@eisti.eu](mailto:fd@eisti.eu) (F. Devin), [dominique.laffly@gmail.com](mailto:dominique.laffly@gmail.com) (D. Laffly), [yl@eisti.eu](mailto:yl@eisti.eu) (Y. Le Nir).

<sup>1</sup> <http://www.eisti.fr>.

<sup>2</sup> This means that they are also automated.

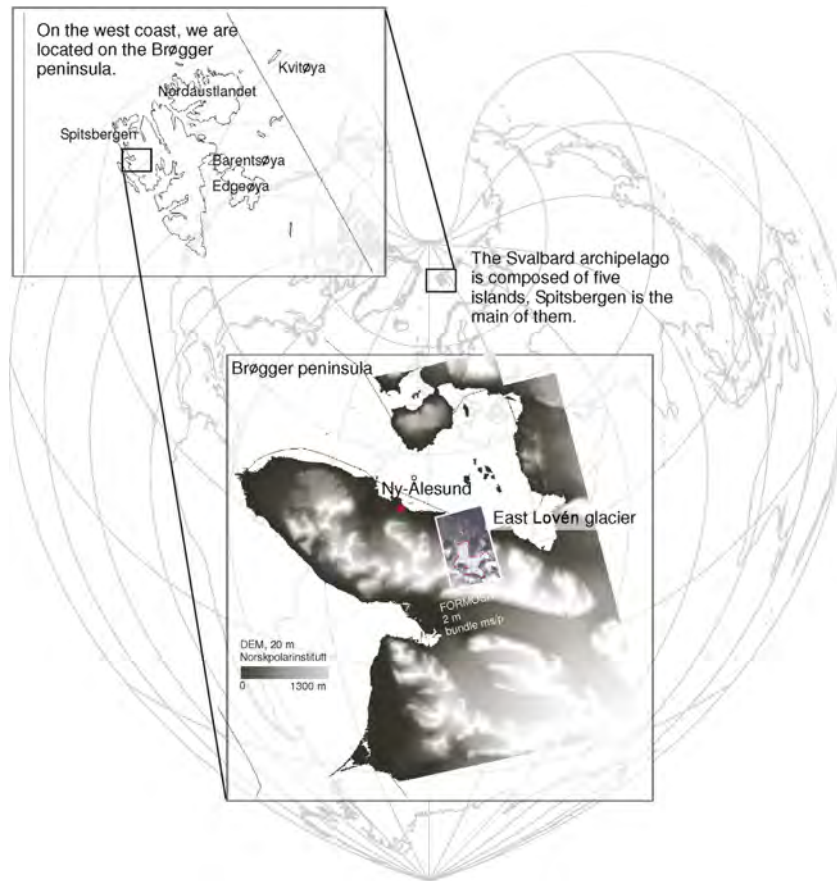


Fig. 1. Location of the East Lovén glacier.

## 2. Objectives

Nowadays, spatial environmental analysis is done using specific tools. These tools may be GIS like ArcGIS, or a simple script or program developed internally. They are not always user-friendly, and do not allow mass processing. Each method of processing has to be done specifically for each group of data.

This gives rise to the following questions: how can a tool be created which can fit the full process of this analysis and which can be re-usable in other similar analyses? How can it be made dynamic enough to adapt to the different possible scenarios? Moreover, how is such a large database to be used?

We will first discuss about the data and the different tasks needed to orthorectify the images in order to define the temporal evolution of the snow cover of the glacier (Section 3). Then, we will propose an architecture model of a standalone tool which can classify a large variety of phenomena avoiding human operation as much as possible. This model is based on cloud computing that we will introduce in Section 4. We will finally present the model and architecture of the application in Section 5.

## 3. Data: processing and disturbances

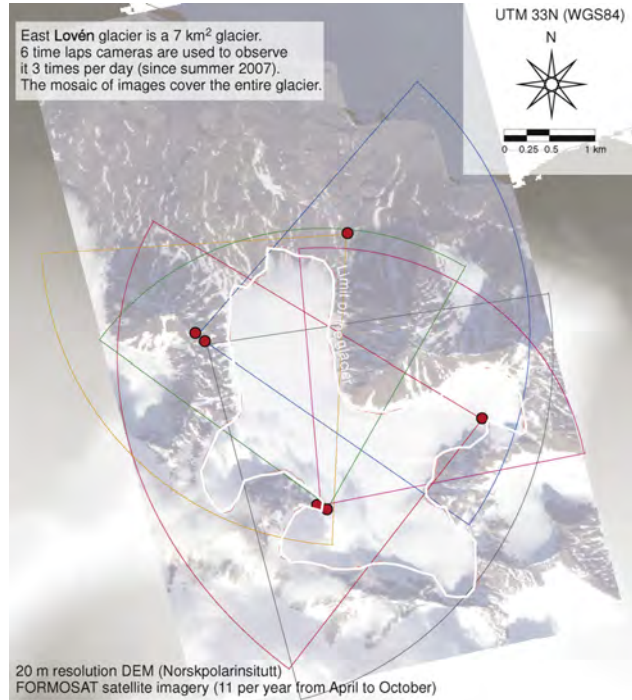
In order to build up a mosaic from the *in situ* pictures projected, the data have to be processed. This process can be split into several atomic tasks: classification, cleaning, projection, dating and cropping. In our situation, cropping is a transformation that combines rotation and translation. Not all the pictures need to be processed

by all the tasks, and the calling order of the different tasks may vary. As the series of the tasks in our model can vary, we only focus on the functionality of the tasks rather than their implementations.

### 3.1. Classification

To identify the different parts of the glacier (snow, ice, rock, etc.) a classification task is needed. There are many algorithms of classification that can be termed as either supervised or unsupervised. A supervised learning algorithm applies the knowledge learnt from the training data to the new data. Whereas an unsupervised learning algorithm splits the data into groups in order to join together data that have the most similarities. Among them, the standard algorithms of geographical satellite-image classification are K-means (Duda et al., 2000; Marroquin and Girosi, 1993), maximum-likelihood (Venkateswarlu and Raju, 1991) and Support Vector Machines (Cortes and Vapnik, 1995; Schölkopf and Smola, 2001).

The classification can be applied at different levels. On the one hand, it can be realized on the map projection reconstructed from the *in situ* pictures. The drawback is that the quality of the image is reduced due to the processing which has been done several times before, so the precision of the classification might be reduced too. Alternatively, it can be realized on the *in situ* pictures. At that state of the process, the pictures still have the same quality, but it means that the result of the classification will have to be processed in the same way as the pictures in order to obtain the map projection.



**Fig. 2.** Cameras, represented by red dots, are placed on the ridges surrounding the glacier. Their different viewing angles provide a full coverage of the glacier. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

### 3.2. Cleaning

Despite the quality of the pictures for the classification, poor weather conditions may partially or totally obstruct the view. As explained before, the cameras are autonomous and subjected to snow, rain and fog. The cleaning task is designed to put aside the pictures which are obstructed by the weather and from which no information can be extracted, or to select pictures from which some information can be extracted.

Fig. 3 is a sample of the different types of disturbance that can occur. The complexity of the task will depend on the type of

disturbance. For example, Fig. 3a–c has a flat histogram indicating little contrast. They can easily be put aside. However, for Fig. 3d–f, it is harder to define if they are usable and if so, how much of each one may be used. Up to now, it has been done manually. With the classification task however, the disturbance is detected as noise or unknown information.

### 3.3. Projection

To go from the *in situ* pictures to a mosaic representing an equivalence of the satellite image, the pictures have to be projected. We use UTM zone 33N projection using the WSG84 datum. The pictures are then taken from the different cameras at the same time and mapped together.

Due to the lack of information about the calibration and the shooting conditions of the cameras, classic projection algorithms like orthorectification cannot be used. Such errors in the projection result have already been described (Hinkler et al., 2002) and are inherent to ground based cameras acquiring oblique view pictures of the landscape. For the *in situ* pictures, the foreground and the background cannot be adjusted in a uniform manner. The proposed alternative is to use a rubber sheeting geometrical correction as it is a procedure to adjust coverage features in a non-uniform manner. The pictures are linearly interpolated using Delaunay triangulation. This method requires the definition of many reference points (Laffly et al., 2012) which are in common between the satellite and *in situ* image. An example of the result is given in Fig. 4.

The results of all the projected pictures acquired at the same time and date are then assembled. They provide a complete coverage of the whole glacier as presented in Fig. 5 for one particular dataset.

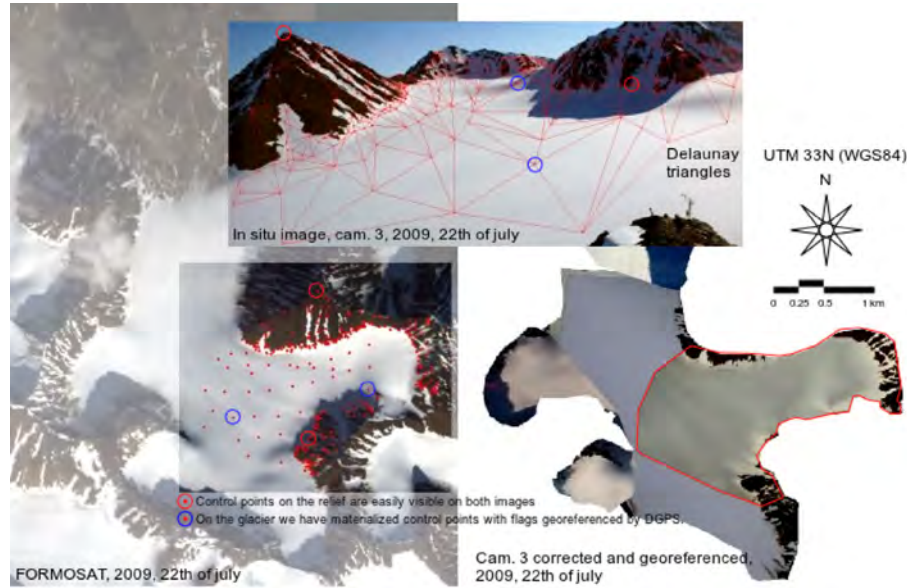
### 3.4. Dating

To combine the pictures into one map, they have to be taken at the same time as the objective is to see the evolution of the snow cover in the timeline.

A microcontroller has been programmed to define when to switch on the camera, switch it on, focus, trigger the picture capture sequence and switch off the camera, all of which happens three times a day (Laffly et al., 2012).



**Fig. 3.** Different types of disturbance.



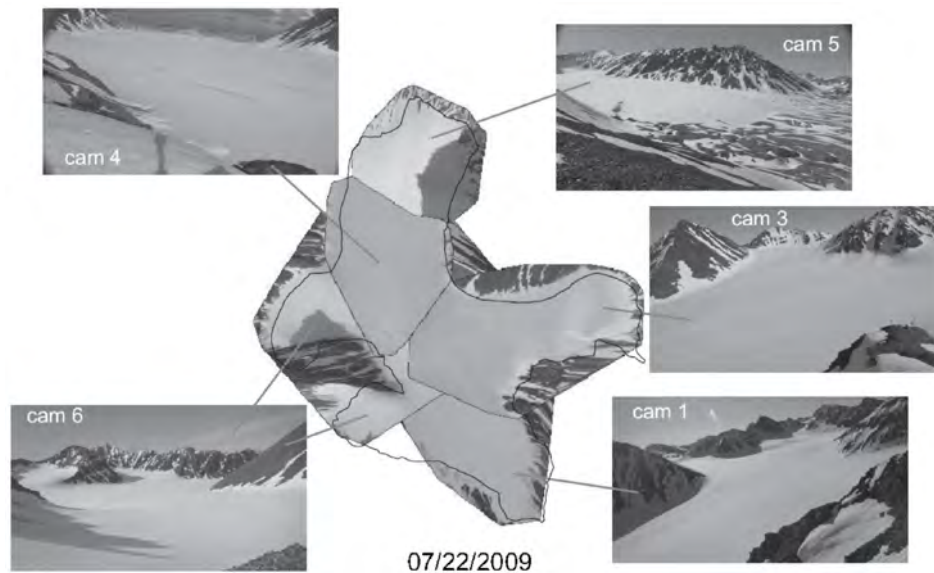
**Fig. 4.** Delaunay triangulation and rubber sheeting. At the top, the *in situ* image with the Delaunay triangulation. At the bottom left, the FORMOSAT image, with the control points (CP). CP on the relief are easily visible on both images (red circles). On the glacier, we have materialized CP with 2 m × 2 m orange flags georeferenced by Differential Global Positioning System with an accuracy of 10 cm (blue circles). At the bottom right, the image projected with the rubber-sheeting geometrical correction in the mosaic. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

As the cameras are autonomous, the life of the internal battery which controls the real time clock varies according to the climate conditions (ranging from one to three months). When the internal battery is discharged, the date information in the EXIF (exchangeable image file format) header is unknown. Not only is the date information lost but also the microcontroller resets. The reset changes the frequency of the snapshots from three pictures every 4 h to six pictures every 2 h. On top of that, bad weather conditions disturb the sequence of the snapshots (e.g., the camera cannot focus because of snow on the lens so the picture is not taken).

This dating task calculates the original date and time of the picture.

### 3.5. Cropping

In order to project the pictures on the plan, many CP have been defined between the *in situ* and satellite image. These reference points have been defined once for the whole project. At each maintenance operation due to the discharge of the camera battery, the camera is removed from its stand, opened to change the battery,



**Fig. 5.** Mosaic and field of view of each camera (Laffly et al., 2012).



Fig. 6. Variation of view between 2007 and 2008.

and then put back on the stand. This manipulation causes variations in the picture's view (example in Fig. 6).

As all the control points are only known for the reference *in situ* picture, one of the tasks is to determine if the pictures are off-centered from the reference picture, and then restore the view. As the cameras are fixed on the stand, the possible variation is a rotation and/or a translation. So, the correction is made by a two-dimensional polynomial.

The control points for the cropping can be selected by the user, or they can be defined automatically with auto-correlation methods.

With all these tasks, it is possible to build up a mosaic in map projection from the *in situ* pictures to define the temporal evolution of the snow cover. One example of a sequence of tasks can be:

1. *crop* all the pictures which were taken at the same time
2. *project* them, and then build up a mosaic
3. *classify* the different phenomena (ice, snow, etc.) existing in the glacier

Another example could be: *clean*, *date*, *crop*, *classify* and *project*. Of course, the sequence will not be the same for all the pictures. It will depend on the pictures: if the date is lost we have to use the dating task, if the weather was bad we have to clean the picture if it is usable.

All these choices can be made since all the data have been previously annotated to add semantic meaning. As the model knows the specificity of the data and the requirements of the tasks, it can decide the appropriate processing chain order.

We want to be able to increase the number of pictures and tasks without limitation. For processing a lot of data that can scale dynamically, the concept of cloud computing is very appropriate, as we will explain in the next section.

#### 4. Cloud computing

Cloud computing is the use of computing resources (computing power, data storage, hardware, software, etc.) that are delivered as a service over a network (typically the Internet) (Syntec Numérique, 2012).

The National Institute of Standards and Technology (Mell and Grance, 2011) gives several essential characteristics of cloud computing:

**On-demand self-service:** The user can supply any computing capabilities (server time, network storage) as needed, without requiring human interaction with the provider.

**Broad network access:** The services are available over the network and accessed through standard mechanisms for any kind of

client platforms (e.g., mobile phones, tablets, laptops, and workstations).

**Resource pooling:** The resources are pooled to serve multiple users. They are dynamically assigned and reassigned according to consumer demand. The user has no information about where the resources are, how many resources are requested or who else is using them.

**Scalability:** The resources can be elastically provisioned and released to adjust to the demand. For the user, the resources appear to be unlimited and can be appropriated in any quantity at any time.

Cloud computing provides several advantages like ease of access, lightness of application, scalability, etc. Moreover large amounts of data can be loaded in the cloud computing system. It also gives the opportunity to add or remove functionalities to fit the user's needs. Finally, it supports not only the increase of users connected to the application, but also the computing capacity according to the needs.

##### 4.1. Deployment models

In a cloud architecture, we have to deploy different services (SaaS, PaaS, IaaS) as described by Furht (2010), Mell and Grance (2011) and Syntec Numérique (2012). This deployment can be realized in three different infrastructures:

**Private cloud:** The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It is owned, managed, and operated by the organization.

**Public cloud:** The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of these.

**Hybrid cloud:** The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private and public) that enables data and application portability.

The choice of the deployment model for an application (Syntec Numérique, 2012) is made according to its level of:

- *Specificity, genericity:* ability of an application to fit other use case (genericity) or only the one it has been created for (specificity);
- *Criticality:* importance of several aspects like its interactions, its dependences, the risk involved if it breaks down, etc.;
- *Scalability:* as defined above.

The need of a high level of scalability leads to the use of cloud computing (which is by definition the most scalable model). In the case where the application must be scalable, the choice between public and private cloud is made according to the level of specificity or criticality. If the level of specificity or criticality is low, then a public cloud is a good solution. Otherwise, the private cloud is more appropriate. If the application has a low level of specificity or criticality and of scalability, then virtualization which consists of deploying a full environment on a distant machine is suitable. Otherwise, the application is deployed in the conventional way. All this is summarized in Fig. 7.

The scalability of the cloud enables to adjust the resources in line with the demand. It gives to the tasks the possibility of adapting to the flow of data to process (pictures in our case). For example, in the workflow, we will always have to project and build a mosaic from the data. We will not always need to date the pictures, however. The structure of the cloud will automatically increase the number of projections and mosaic services by duplicating them, while it will



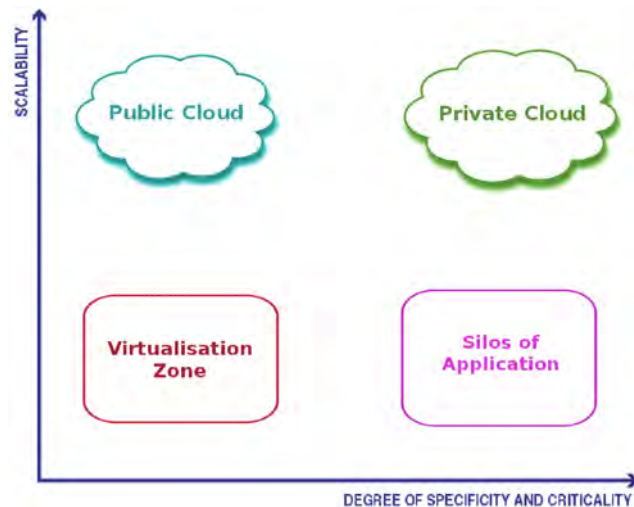


Fig. 7. Deployment strategy.

not be necessary for the dating service. The increase of data is also managed with the scalability of the model since the amount or the size of the pictures can vary. The level of specificity and criticality of the tasks varies from one to another, without being extreme. Therefore, any cloud computer infrastructure (private, public or hybrid) is appropriate. For example, cropping service is not specific and already exists in a public cloud whereas dating service is more specific to our situation so it can be deployed on a public or private cloud. In our case, we have created a private cloud with some computers that are not dedicated to it. Like many computers, their load average is very low and many tasks can be executed in parallel without performance consequences for the user. We might increase our cloud by making a hybrid cloud linking our private cloud with a public cloud.

#### 4.2. Web Services

A Web Service (WS) is usually seen as an application accessible to other applications over the Web (Alonso et al., 2003).

The World Wide Web consortium defines it as: “a software application identified by a URI (Uniform Resource Identifier), whose interfaces and binding are capable of being defined, described and discovered by XML (Extensible Markup Language) artifacts and supports direct interactions with other software applications using XML based messages via Internet-based protocols” (W3C, 2002).

Now that we have defined the image processing through the different tasks and what cloud computing is, we are going to present a model and architecture to define the temporal evolution of the snow cover.

### 5. Model and architecture

As explained before, in a cloud environment, the different tasks can be accessed over the network, using Web services, for example.

In order to use a WS over the Web, the user has to discover its existence and the description and definition of the different functionalities it offers. The user will not know how the method is implemented but only what it does. All the interactions between the application and the user consist of messages over the Web. The user can be a human or another application. It enables the linking of WS together in order to achieve a more complex processing chain.

As explained in Section 3, we build up a mosaic from the *in situ* pictures projected to define the temporal evolution of the snow

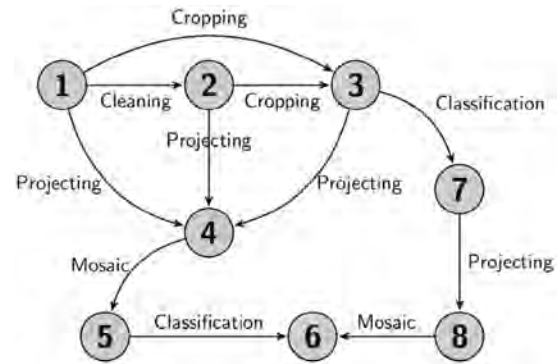


Fig. 8. Sample of workflow of the image processing. The images are the vertices and the tasks are the edges. Vertex 1 is the initial images and vertex 6 is the result of the image processing. According to the state of the pictures (off-centered or not, cleaned or not, etc.) and according to the services available, different workflows are possible.

cover. The data processing can be split into several tasks. All the tasks are “atomic” (processes which consist of one elementary action). The composition of tasks makes it possible to adapt to different configurations. The process is therefore generic. In our model, we have decided to implement and deploy each task as a WS.

#### 5.1. Scheduler

Lin (2012) defines a workflow as follows: “a workflow is composed of a set of tasks which are organized according to a logical order, in order to realize a global task, complex and relevant on a set of data”.

By splitting the processes into multiple tasks, we have to schedule them, with a workflow, in order to obtain the temporal evolution of the snow cover. A workflow composed only of WS is called a WS composition.

Fig. 8 is a representation of an image processing workflow. The images are the vertices and the tasks are the edges. State number 1 is the initial images. State number 6 is the result of the image processing which is the classification of the glacier phenomena. All the paths correspond to the different ways of obtaining the classification.

If the pictures had not been disturbed, the path would have been {1, 4, 5, 6}. The disturbances lengthen the path, by adding tasks to the processing sequence.

In this case of image processing, there is not only one workflow. It will depend on the state of the pictures. If the picture has no date information, the process will go through the dating task. The workflow can also vary according to the services available. There can be different services of classification, supervised or unsupervised, available on the cloud. The services may use *in situ* pictures or mosaic images. All the services may be accessible or not, may improve according to new algorithms, may be created. Because of these variations, many workflows exist.

Two approaches exist for WS composition (Contes, 2011; Peltz, 2003):

**Orchestration of services** describes a centralized vision of the service composition. A central entity manages the order of the invocation of the different services according to a defined workflow. It is a static WS composition.

It requires an orchestra usually represented by a state diagram as in process models like *XML Process Definition Language* (XPDL),

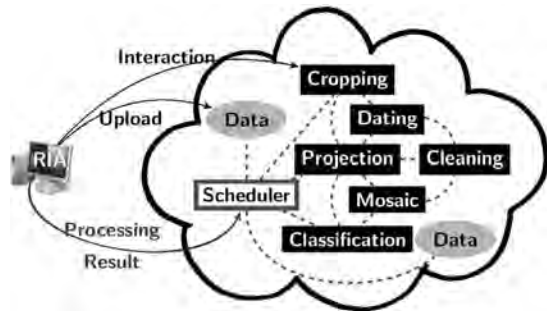


Fig. 9. Architecture of the application.

Business Process Execution Language (BPEL) or Business Process Modeling Notation (BPMN) (Shapiro, 2002).

**Choreography of services** describes, from a general point of view, how the services interact with each other in order to realize a common goal. It is a dynamic WS composition.

The choreography tries to suppress this explicit orchestra and to replace it with automatic tasks. It is a well-known problem from Service Oriented Architecture (Karande and Karande, 2011) but many semi-automatic methods have been tried. For example, the semantic we added to data and services can be used to create the appropriate choreography (Klusch and Gerber, 2005). Other methods can be used as described in Bourdon (2007) or Rao and Su (2005).

As the choreography is a dynamic composition, defining a scheduler of this type will provide the dynamic aspect of the model. Indeed, the scheduler will adapt the workflow to the existing services and the states of the picture.

In our case, we use the semantic annotations added on the data and the services to create the choreography. The semantic allows us to discover the different workflows. We can evaluate the quality of each workflow comparing for example the ratio of well classified pictures.

## 5.2. Summary of the model architecture

The image tasks are WS which are deployed on the cloud computing system. As explained in Section 4.1, according to our needs, the hybrid cloud seems the most appropriate one. For some of the services, the user will have to interact with it (e.g., a supervised classification, a cropping task with control points selected by the user). All the pictures are uploaded and stored in the cloud as well. The scheduler manages the interaction between the data and the services.

All this is summarized in Fig. 9.

## 6. Conclusion and perspective

In order to analyze the temporal evolution of the snow cover of a glacier, we use *in situ* pictures from cameras placed around the glacier. To extract information, the pictures must be processed in order to obtain a temporal map of the different states of the glacier. In this paper, we have presented a dynamic and generic cloud computing model for a standalone tool which processes the *in situ* pictures. The dynamicity of the model comes from the use of cloud computing and the WS choreography. Cloud computing gives the modularity in WS (adds, removes or changes it), and the choreography gives the possibility of dynamically adapting to the services accessible. The genericity of the model comes from the genericity of each WS. As they are “atomic”, they can fit other situations and other types of data.

To date, we have already implemented the services and our model has been used to classify a large amount of data from 2008 and 2009, representing 18Gb of pictures.

One prospect is to improve the dynamic scheduler with the use of a concurrent constraint logic programming. In future, we could analyze the impacts of the changing of the analyzed site. For example, the analysis of the same type of phenomena in other places with different weather conditions (e.g., Pyreneas’s glacier where the main constraint is the high rate of humidity) or other types of phenomena that can also be analyzed with *in situ* cameras (e.g., soil erosion due to intensive farming in Asia). Our model is built so that specific services can be easily added or substituted in order to adapt to the particularity of each site. Based on this study, we could develop a new scheduler that could automatically choose the appropriate services taking into consideration the environment.

## Acknowledgements

We want to thank the HSF team for the access to the database of the program, for their expertise, and for the time they took to explain the specifics of the issue.

We would also like to thank the anonymous referee for their constructive remarks which have greatly improved this paper.

## References

- Alonso, G., Casati, F., Kuno, H., Machiraju, V., 2003. *Web Services: Concepts, Architectures and Applications*. Springer Verlag.
- Bernard, E., 2011. Les dynamiques spatio-temporelles d’un petit hydrosystème arctique: approche nivo-glaciologique dans un contexte de changement climatique contemporain (bassin du glacier Austre Lovén, Spitsberg, 79N). Université de Franche-Comté (Ph.D. Thesis).
- Bourdon, J., 2007. June. Multi-agent systems for the automatic composition of semantic web services in dynamic environments. University of Saint-Etienne (Master’s thesis).
- Contes, V.L., 2011, December. Une approche à composant pour l’orchestration de services à large échelle. Université de Nice – Sophia Antipolis (Ph.D. Thesis).
- Corripio, J.G., 2004. *Snow surface albedo estimation using terrestrial photography*. *International Journal of Remote Sensing* 23 (24), 5705–5729.
- Cortes, C., Vapnik, V., 1995. Support-vector networks. *Machine Learning* 20 (September), 273–297.
- Duda, R.O., Hart, P.E., Stork, D.G., 2000. *Pattern Classification*, 2nd ed. John Wiley & Sons, USA.
- Furht, B., 2010. *Cloud Computing Fundamentals*. Springer, US.
- Hinkler, J., Petersen, S.B., Rasch, M., Hansen, B.U., 2002. Automatic snow cover monitoring at high temporal and spatial resolution, using images taken by a standard digital camera. *International Journal of Remote Sensing* 23 (21), 4669–4682.
- Karande, A., Karande, M., Meshram, B.B., 2011. Choreography and orchestration using business process execution language for SOA with Web services). *International Journal of Computer Science Issues* 8 (2), 224–232.
- Klusch, M., Gerber, A., 2005. Proceedings of the 1st Int. AAAI Fall Symposium on Agents and the Semantic Web, pp. 55–62.
- Laffly, D., Bernard, E., Friedt, J.-M., Martin, G., Tolle, F., Marlin, C., Griselin, M., 2012. High temporal resolution monitoring of snow cover using oblique view ground-based pictures. *Polar Record* 48, 11–16.
- Lin, Y., 2012. Méthodologie et composants pour la mise en oeuvre de workflows scientifiques. Université de Montpellier II (Ph.D. Thesis).
- Marroquin, J.L., Girosi, F., 1993. Some extensions of the K-means algorithm for image segmentation and pattern classification, MIT Technical Report, AI Memo No. 1390.
- Mell, P., Grance, T., September 2011. The NIST definition of cloud computing. NIST Special Publication 800-145.
- Newbery, K.B., Southwell, C., January 2009. An automated camera system for remote monitoring in polar environments. *Cold Regions Science and Technology* 55, 47–51.
- Peltz, C., 2003. Web services orchestration and choreography. *Computer* 36(October(10)), 46–52.
- Rao, J., Su, X., 2005. A survey of automated web service composition methods. *Semantic Web Services and Web Process Composition – Lecture Notes in Computer Science Volume 3387*, 43–54.
- Schölkopf, B., Smola, A., 2001. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. Cambridge, MA: MIT Press.
- Shapiro, R., 2002, September. A Comparison of XPD, BPML, and BPEL4WS. URL: <http://xml.coverpages.org/Shapiro-XPDL.pdf>
- Syntec Numérique, 2012. Livre blanc cloud computing “Nouveaux modèles !”.
- Venkateswarlu, N., Raju, P., 1991. Three stage ML classifier. *Pattern Recognition* 24 (11), 1113–1116.
- W3C, 2002, October. Web Services Description Requirements. URL: <http://www.w3.org/TR/ws-desc-reqs/>

### 8.3 Conclusion partielle

Afin de traiter automatiquement l'ensemble des données acquises depuis les appareils autonomes, nous avons mis en place un modèle de *cloud* où, avec l'ensemble des traitements déployés en tant que WS, un ordonnanceur permet d'appeler les bons traitements sur les bonnes photos et dans un ordre adéquat.

Grâce à ce modèle, nous pouvons obtenir un outil dynamique et générique. Dynamique car l'utilisation du *cloud computing* apporte la modularité dans l'utilisation des WS (ajout, suppression, mise-à-jour) et l'ordonnanceur permet de s'adapter à la volée aux services accessibles au moment de l'exécution. Générique car le modèle étant formé de plusieurs services effectuant le traitement, ces services sont "atomiques" et peuvent s'adapter à l'analyse d'autres paysages. En effet, ce modèle a été conçu initialement pour le traitement des photos *in situ* afin de cartographier la couverture neigeuse du glacier du Lovén Est. Cependant, dans une configuration semblable, c'est-à-dire où la zone à analyser est entourée d'appareils photo qui prennent régulièrement des clichés, et pour le même objectif de cartographier l'évolution des différents composants du paysage, le modèle et les services peuvent être directement utilisés. Si cependant d'autres traitements sont souhaités, comme par exemple la définition des vecteurs d'évolution d'un front de glacier, seuls les WS sont à développer et intégrer dans le *cloud*. Évidemment, changer le traitement d'analyse environnementale souhaité correspond à définir un tout autre ordonnancement des services. La composition de services reste un point clé de la dynamique du modèle.

## Chapitre 9

# L'orchestration des services

Logic will get you from A to B.  
Imagination will take you everywhere.

---

*Albert Einstein*

### 9.1 Introduction partielle

Les contraintes sur les données acquises afin d'analyser la couverture neigeuse d'un glacier ainsi que son évolution sont multiples et irrégulières (chapitre 5). Comme nous avons pu le voir à travers l'article précédent [Ranisavljević et al., 2014a], mettre en place une chaîne de traitement qui permet de traiter toutes les données acquises correspond à définir un ordonnanceur de services qui peut s'adapter aux différentes perturbations des photos.

Parmi les différentes méthodes de composition de services (chapitre 6), nous avons fait le choix de partir d'une composition au format BPEL, et de le diversifier et le rendre plus dynamique et adaptable aux différents scénarios par la sémantique. Les WS étant développés en SOAP avec un descriptif WSDL pour des raisons de quantité de données à transférer à travers les requêtes, l'ajout de la sémantique aux différents services a été fait à travers les annotations en SAWSDL.

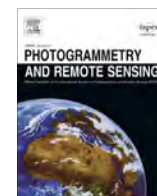
Le modèle ontologique utilisé pour les annotations sémantiques correspond à la description des différents WS définis pour le traitement de l'analyse environnementale. Il définit les familles de types de traitement (services de classification, de mosaïque, etc) ainsi que les différents types de données utilisés en entrée et sortie de chacun des services. Ainsi, par des requêtes spécifiques sur cette ontologie, il est possible de définir ou diversifier des *workflow* en fonction du traitement souhaité.

### 9.2 Article "*Semantic orchestration of image processing services for environmental analysis*"



Contents lists available at SciVerse ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: [www.elsevier.com/locate/isprsjprs](http://www.elsevier.com/locate/isprsjprs)

# Semantic orchestration of image processing services for environmental analysis



Élisabeth Ranisavljević<sup>a,\*</sup>, Florent Devin<sup>a</sup>, Dominique Laffly<sup>b</sup>, Yannick Le Nir<sup>a</sup>

<sup>a</sup> École Internationale des Sciences du Traitement de l'Information, 2 boulevard Lucien Favre, CS 77563, 64 075 PAU CEDEX, France

<sup>b</sup> University of Toulouse, GEODE UMR 5602 CNRS, Maison de la recherche, 5 allées Antonio Machado, 31 058 TOULOUSE CEDEX, France

## ARTICLE INFO

### Article history:

Available online 18 July 2013

### Keywords:

Environment  
Image processing  
Snow ice  
Web service  
Orchestration  
Semantic

## ABSTRACT

In order to analyze environmental dynamics, a major process is the classification of the different phenomena of the site (e.g. ice and snow for a glacier). When using *in situ* pictures, this classification requires data pre-processing. Not all the pictures need the same sequence of processes depending on the disturbances. Until now, these sequences have been done manually, which restricts the processing of large amount of data.

In this paper, we present how to realize a semantic orchestration to automate the sequencing for the analysis. It combines two advantages: solving the problem of the amount of processing, and diversifying the possibilities in the data processing. We define a BPEL description to express the sequences. This BPEL uses some web services to run the data processing. Each web service is semantically annotated using an ontology of image processing. The dynamic modification of the BPEL is done using SPARQL queries on these annotated web services.

The results obtained by a prototype implementing this method validate the construction of the different workflows that can be applied to a large number of pictures.

© 2013 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS) Published by Elsevier B.V. All rights reserved.

## 1. Introduction

### 1.1. Context

Analyzing landscapes, its dynamics and environmental evolutions require regular data from the sites. In the following, we will use as an example the analysis of the East Loven glacier mass balance in Spitsbergen (78°N, 12°E, Svalbard, Norway). Several readings must be done at different levels (Bernard, 2011). One level would be in the field, for example reading temperature, snow density and water flow. The other level would be a distance reading, like getting a global view of the glacier (snow/ice blanket) and its evolution with remote sensing control. The environmental analysis is obtained by linking all these pieces of information together (temperature, water balance, classification of the different glacier states, etc.).

Ideally, satellite images should be used for the classification. Nevertheless, these images are not always workable due to poor weather conditions (including common heavy cloud cover at polar

latitudes). Moreover, they are not always affordable. Besides, fast events like floods or blankets of snow are missed by satellite based studies, since the slowest sampling rate is unable to observe them (Bernard, 2011). In order to complement satellite imagery (FORMOSAT), a set of ground based autonomous automated digital cameras have been installed in the glacier, as part of the scientific program *Hydro Sensor Flow* supervised by Corripio (2004), Hinkler et al. (2002), Laffly et al. (2012), Newbery and Southwell (2009). The instruments were placed on the ridges surrounding the East Loven basin in order to cover most of the glacier basin with the different view angles. “Each instrument is based on a COTS (commercial off-the-shelf) digital camera (either Leica D-Lux4 or Panasonic Lumix LX3 – the latter being packaged in a stronger aluminum case is more difficult to modify), selected for its lens optics quality, high sensor resolution (10 Mpixel sensors), and data storage on Secure Digital (SD) mass storage memory card. Indeed, such a mass storage provides both the bandwidth needed to keep the camera powered for as short a duration as possible (in order to reduce global power consumption) while requiring few enough signals for the card holder to be located away from the camera itself, accessible to the user while keeping the camera in an air tight enclosure.” – (Laffly et al., 2012). An example of such a concept is given in Fig. 1. The cameras have been programmed to take 3 pictures a day (08:00, 12:00 and 16:00) in order to follow the

\* Corresponding author.

E-mail addresses: [erc@eisti.eu](mailto:erc@eisti.eu) (É. Ranisavljević), [fd@eisti.eu](mailto:fd@eisti.eu) (F. Devin), [dominique.laffly@gmail.com](mailto:dominique.laffly@gmail.com) (D. Laffly), [yl@eisti.eu](mailto:yl@eisti.eu) (Y. Le Nir).

URL: <http://www.eisti.fr> (É. Ranisavljević).

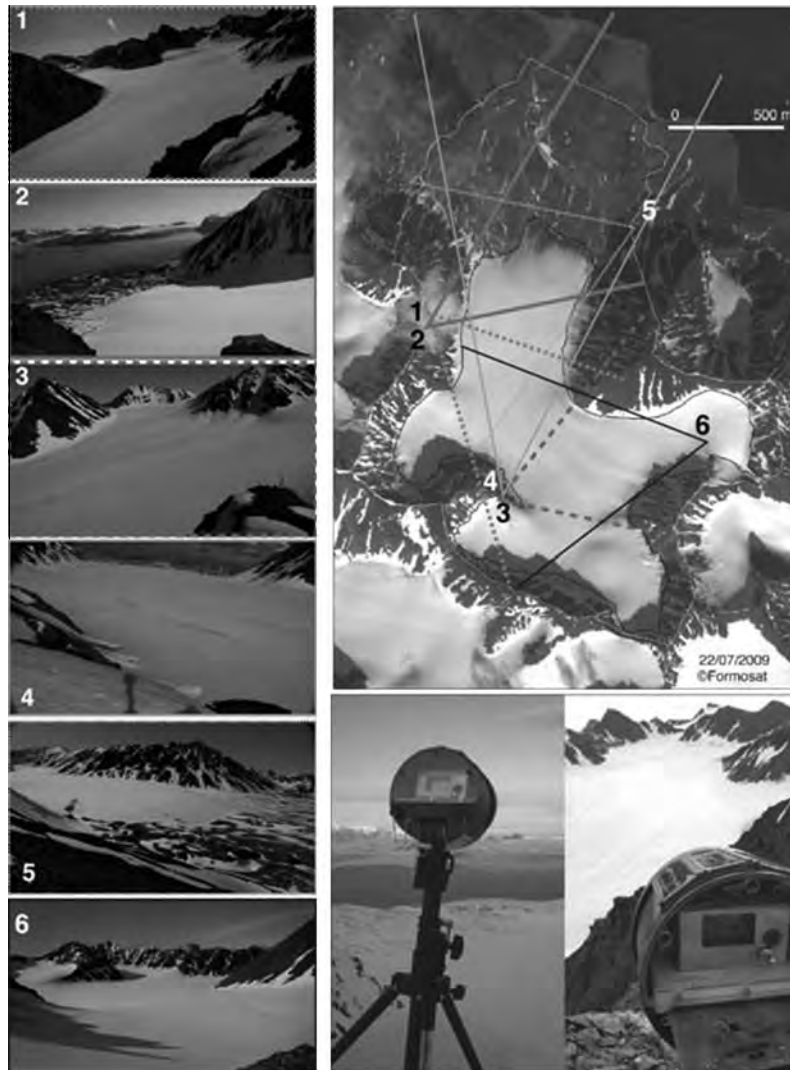


Fig. 1. Ground based autonomous automated digital cameras for daily reading (Laffly et al., 2012). Left: the *in situ* pictures. Top right: position of the cameras and their view angle on the satellite image. Bottom right: experimental setup, in which a digital camera is packaged in a waterproof, air tight enclosure.

spatio-temporal evolution of the glacier regularly. Taking more pictures should provide useful information, but it fills quicker the memory card. This implies to collect the data more often. Taking less pictures reduces the probability to detect fast events. These pictures form a huge database, but are disturbed by:

- weather conditions: water drop, fog, ice on the pictures, etc.;
- electronic deficiency: more or less pictures taken a day;
- physical constraints on the camera: geometry variations on the pictures.

To use these pictures in spite of the disturbances, several tasks must be performed. The goal of all the tasks on the pictures is to create a mosaic to the photos as an alternative of the satellite images. The aim is also to classify the different phenomena, in order to analyze the evolution of the glacier phenomena (ice, snow, etc.). In order to get the same view, all the pictures have to go through several tasks.

### 1.2. Image processing

We can distinguish 2 types of tasks: the ones due to unpredictable constraints of the site (e.g. the weather condition) and the

ones due to the transformation of the *in situ* pictures to an alternative of the satellite view.

Among the tasks due to unpredictable constraints of the site, we have: the usability and cleaning, the dating and the cropping of the pictures.

- **Usability and cleaning:** The tasks of usability of pictures detect the pictures which are too greatly disturbed by poor weather conditions like fog, snow or water on the lens. Some perturbations can be cleaned, but others make the pictures meaningless. The goal of this task is to clean the picture if possible and to remove the pictures which do not give any information about the glacier from the set of pictures.
- **Dating:** One of the reasons for maintenance operation is the discharge of the camera's internal battery. In order to take three pictures a day, a microcontroller has been programmed to capture 3 images a day. This microcontroller allows the camera 20 s to turn on, to focus, to take the picture and to save it on the memory card. Due to poor weather conditions, there are some situations where the microcontroller is "asleep" for several days. Moreover, if the camera does not succeed in taking the picture in the 20 s (e.g. if there is snow on the lens and the camera cannot focus), the picture is not taken. As the cameras are

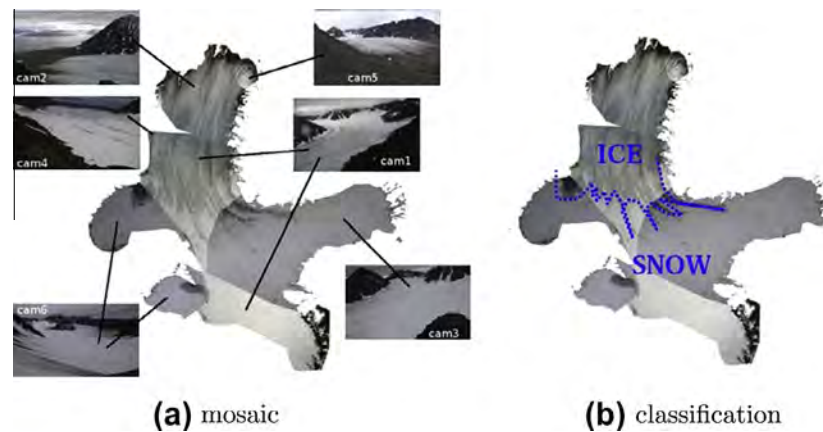


Fig. 2. Glacier reconstruction and classification.

autonomous in extreme conditions, maintenance operation has to be done approximately every 6 months to collect the photos and to repair the cameras which have broken down. On top of that, the microcontroller resets under the electronic deficiency which changes the snapshot frequency from 3 pictures every 4 h to 6 pictures every 2 h. The date information set on the picture when it is taken is very important to place it in the timeline. It has been noticed that the internal battery lasts from 1 to 3 months. After that time, the pictures are taken, but with no information about the date. There are many pictures that do not have any date. Due to the internal battery leak and the microcontroller deficiencies, we have to find the date of pictures when needed. The goal of the dating task is to find the date of the snapping when it has been lost, by calculating it whenever possible.

- **Cropping:** The maintenance operation requires taking the camera off the stick in order to manipulate it, and then putting it back. This manipulation causes the variation of the picture's view. For the projection, one of the transforming tasks, we have defined several Ground Control Points (GCP) which are in common between a reference picture of each camera and the satellite image. In order to be able to use these GCP, we need to have only well-composed pictures. The goal of this cropping task is to detect all the pictures which are off-centre from the reference picture and to restore the view.

After completing the necessary tasks on the pictures according to their state, the set of pictures is usable for the following tasks. In order to obtain a view similar to the satellite view, we have defined two tasks for all the pictures: their projection on a plan, and the constitution of a mosaic with all the projected pictures taken at the same time.

As all the pictures have been taken from the glacier, they all have a strong isometric. In order to have a global view, the picture must be projected on a plan. The GCP gives the opportunity to use a rubber-sheeting geometrical correction model to project the pictures on the plan (Laffly et al., 2012). We use UTM zone 33 North projection in relation with WGS84 DATUM. When all the pictures have been projected, a mosaic of the glacier can be made. As explained before, the set of cameras covers the whole glacier. Ideally, the reconstruction of the glacier should be performed with the pictures taken at the same date. As the meteorological disturbance impacts the microcontroller and the camera, some pictures might be missing in the timeline. Besides, some zones are visible from several cameras. The task of the mosaic constitution must take into consideration a set of pictures which match in the timeline, and it also should make a choice between the different pictures which

show the same zone. An example of the mosaic result is given in Fig. 2a.

After processing all the pictures through these different tasks, all the pictures from the different cameras have been transformed to a set of pictures representing the glacier at the different dates.

The last task is the classification of the different phenomena of the glacier (Fig. 2b). This task can be done at different levels: before the projection on the individual pictures or after the projection on the mosaic pictures. If the classification is done before the projection, the polygon of the classification will have to be transformed the same way as the picture in order to reconstruct the classification on the mosaic too. If the classification is done after the projection, its precision can be reduced due to the deterioration of the picture's quality.

Until now, all these tasks have been done manually, using several different specific tools. It has been done for over 30,000 pictures of 4MB each, taken during the last 4 years. We want to build a unique, generic and dynamic tool, which can process any pictures from any site.

As the whole image processing is complex, every task is implemented as a specific module. The full process is obtained by composing these modules. In order to have a dynamic tool, we use the dynamic feature of the web by deploying the modules as web services (WS). As image processing is encapsulated in a WS, it acts as a black box, so we can easily improve a service by changing the black box for a better one (more accurate, more efficient, less usage memory, etc.). The point is how we can efficiently use each WS to achieve the study. As each image processing is done by web services, we have to run the services in a specific order. There is no need to systematically run all processing on all images. For example, it is not a good idea to do cropping if the image is not usable (due to weather conditions for example). Or, we do not need to run the dating process on all images. There are many cases to which this applies. So for each image we have to introduce a specific way to do the processing. We must provide a scheduler to order the call of web services. As we will see in the following section, this scheduler can be realized in a static, or in a dynamic way. In our situation, as we cannot predict all situations, we cannot define a static scheduler which will cover all the possible sequences of services. So a dynamic scheduler is the most appropriate.

The rest of the paper is structured as follows. We present the necessary preliminaries on web service orchestration in Section 2. These preliminaries contain the description of orchestration (2.2), web service (2.4), ontology (2.5) and a method for web service orchestration using the semantic. After that, we give an example of web service orchestration deployment applied on image processing for environmental analysis in Section 3. Results are

discussed in Sub section 3.4, and we end with conclusions and perspectives.

## 2. From static to dynamic orchestration

### 2.1. Service Oriented Architecture (SOA)

The procedure to develop an application has changed a lot since the beginning of computer science. The reuse of existing pieces of code has been an important motivation in the apparition of objects oriented language. In this paradigm, the developer becomes an architect that will plug altogether different programs to obtain the desired functionalities. Then, a new problem is the communication between programs. They have to understand each other and most of the time, for simplicity and efficiency reasons, programs has to be written in the same language.

To avoid this restriction, an abstraction of this object model appeared, often called component model, to allow programs to interact over many different languages. The components are very similar to objects with the addition of a specific layer to manage interactions between different languages.

With the evolution of networks, this component model has been used to create SOA, where services are programs that can be accessed over a network with specific protocols. The actual cloud computing paradigm follows this evolution. The services from SOA can be seen as part of a cloud architecture, often presented as Software as a Service (SaaS).

### 2.2. Orchestration in SOA

The communication between resources in a heterogeneous system is a very important and difficult task. In SOA, it is often presented as a composition of services that is more or less automatic. We can consider different ways to compose services. The easier way is the manual description of all the interactions that is often called an orchestration. Many specific languages like *XML Process Definition Language (XPDL)*, *Business Process Execution Language (BPEL)* (Karande et al., 2011) or *Business Process Modeling Notation (BPMN)* (White, 2004, 2002) can express the interactions between services but are static and cannot support dynamic evolution of resources, like service replacement or upgrade. A solution is the automatic composition of services without explicit static description (Peltz, 2003). This is usually called choreography, but no real efficient methods offer generic and usable solution. A common approach for this purpose is the use of semantic to describe the services and the interactions between them. The composition can change dynamically if the concepts it needs are present in the semantic description, often presented as ontology. In cloud architecture, the composition is also an essential part and the use of semantic is an interesting challenge that is still a work in progress in many research teams (Peltz, 2003).

In Fig. 3, we present our complete model. Each part will be detailed hereinafter.

### 2.3. BPEL for web services (static composition)

As we can see in Fig. 3, the first row represents the static orchestration in BPEL. This orchestration describes the process in an XML format that will be executed by the application. Some important markers are the following:

- `process`: root element of the process,
- `partnerLink`: linking services with process,
- `invoke`: calling a web service,
- `variables`: declaring variables to manipulate data,
- `assign`: updating the content of variables.

An orchestration in our context can be the following: a set of image projection, a set of mosaic composition and a classification. The result gives a model to classify images' parts between ice and snow. This classification can be used to estimate the glacier mass balance. In BPEL, this workflow can be modeled as in Listing 1.<sup>1</sup>

### 2.4. Web services description

The *Web Service Description Language (WSDL)* is a very commonly used specification from the *World Wide Web Consortium (W3C)* (Chinnici et al., 2007) used for WS. We will focus on three different markers (operation, input and output). These markers can be compared to classical specification of programs and will be used by end users to call the services.

In our particular case, the Listing 2<sup>2</sup> presents a snippet for the service that makes a cropping of an image. Without any further information, this service is only an image transformer.

In order to automatically manipulate the services, we will insert semantic annotation to increase their meaning. This is done using *Semantic Annotation for WSDL (SAWSDL)* markers (Akkiraju and Sapkota, 2007). This markers are expressed as triple (`subject`, `property`, `object`) in the *Resource Description Framework (RDF)* format from W3C. As an example, the classification service will be annotated with the triple (`classification`, `input`, `image`). It means the service of classification uses image as input data.

### 2.5. Semantic annotation

The RDF format is usually implemented in XML. It represents graphs with a collection of triples and gives some predefined properties to express semantic relation between objects and subjects. Subjects represent URI (Unified Resource Identifier), that is, in the case of image processing services, the WSDL description of the services. Objects represent either URI or literals. Literal objects represent explicit values, usually XML types (Peterson et al., 2009) like string or integer. An example of such a description for image processing is given in Listing 3.<sup>3</sup> The expressive power of RDF can be extended using some *Web Ontology Language (OWL)* element. OWL implements all the descriptive logic. It can express more complex relations than RDF, such as equivalence, existential or universal restrictions and symmetries. For example, every image have a non-negative size as shown on Listing 3. We use this formalism to build the semantic graph of image processing concepts and to annotate the image processing services.

### 2.6. First step to choreography: WSDL to RDF mapping

In order to organize dynamic composition, we add semantic description to the services. This is done with a mapping between the WSDL description of the services and the RDF representations. We use the RDF mapping description from W3C (Kopecky, 2007). The implementation made by Jacek Kopecky offers a very simple method to compute the RDF file. The obtained RDF file can then be annotated with semantic information. We have decided to annotate the input, output and operation components of the services. On the Listing 4,<sup>4</sup> we have the semantic description of the cropping service where we can see a link between the service and the ontology description `<sawsdl:modelReference rdf:resource="http://laris.eisti.fr/ImageProcessing#cropping"/>`.

<sup>1</sup> Full BPEL is given at: [http://fd.perso.eisti.fr/ISPRS/orchestration\\_bpel.xml](http://fd.perso.eisti.fr/ISPRS/orchestration_bpel.xml).

<sup>2</sup> Full WSDL is given at: [http://fd.perso.eisti.fr/ISPRS/SOAP\\_Cropping\\_Border\\_Linear.wsdl2](http://fd.perso.eisti.fr/ISPRS/SOAP_Cropping_Border_Linear.wsdl2).

<sup>3</sup> Full ontology is given at: <http://fd.perso.eisti.fr/ISPRS/imageProcessing.owl>.

<sup>4</sup> Full WSDL to RDF is given at: <http://fd.perso.eisti.fr/ISPRS/cropping.rdf>.



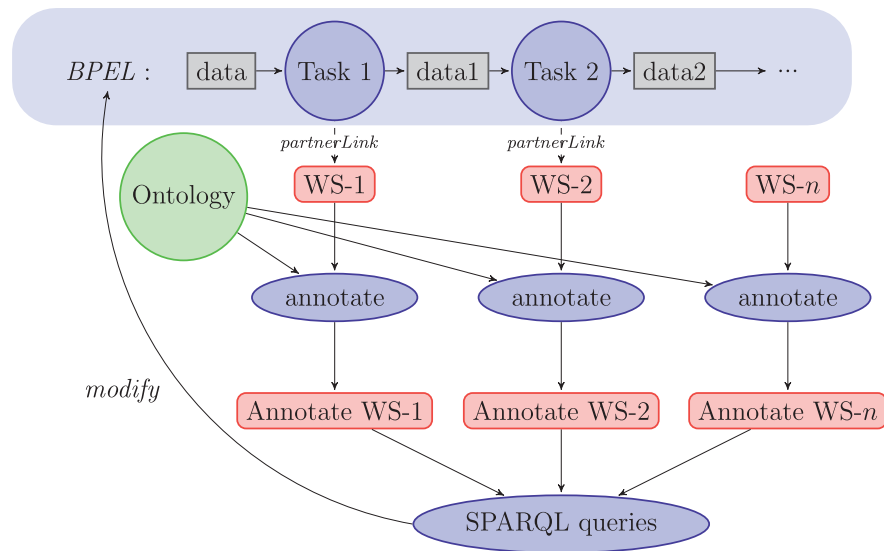


Fig. 3. Complete model of the semantic orchestration.

```

<bpel:process name="ImageProcessing" ...
  <!?partner link definition →
  <bpel:partnerLinks>
    <bpel:partnerLink name="Projection" partnerLinkType="tns:Projection"/>
    <bpel:partnerLink name="PartnerLink" partnerLinkType="tns:Mosaic"></
      bpel:partnerLink>
  </bpel:partnerLinks>
  <!?variable definition →
  <bpel:variables>
    <bpel:variable messageType="..." name="..." />
  </bpel:variables>
  <!?activity definition in the sequence structure →
  <bpel:sequence>
    <bpel:invoke name="Projection" inputVariable="cam1" outputVariable="
      resultCam1"/>
    ...
    <bpel:assign> <bpel:from> ... </bpel:from> <bpel:to variable="
      setOfImage" />
    <bpel:invoke name="Mosaic" inputVariable="setOfImage"
      outputVariable="ImageComplete"/>
  </bpel:sequence>
</bpel:process>

```

Listing 1. BPEL of the projection and mosaic of the pictures.

```

...
<wsdl2:interface name="ServiceInterface">
  <wsdl2:operation name="cropping" style="http://www.w3.org/ns/wsdl/style/
    rpc" wrpc:signature="cropping #in return #return " pattern="http://
    www.w3.org/ns/wsdl/in-out">
    <wsdl2:input element="ns:cropping" wsaw:Action="urn:cropping"/>
    <wsdl2:output element="ns:croppingResponse" wsaw:Action="urn:
      croppingResponse"/>
  </wsdl2:operation>
</wsdl2:interface>
...

```

Listing 2. WSDL for cropping process.

### 2.7. Last move to choreography: semantic queries

Using this semantic description, we can do semantic queries. These queries can retrieve information to dynamically create, modify or replace compositions. There are several techniques to implement queries on RDF graphs. We use the *SPARQL Protocol And RDF*

*Query Language* (SPARQL) (Prud'hommeaux and Seaborne, 2008), that uses a SQL like syntax. With this language, we can express semantic queries in order to dynamically compose services. An example of such a query is given in Listing 5, with a query that retrieves all the operations having an image representing a satellite view as input.

```

...
<owl:DatatypeProperty rdf:about="http://laris.eisti.fr/ImageProcessing#size
">
  <rdfs:domain rdf:resource="http://laris.eisti.fr/ImageProcessing#Image"/>
  <rdfs:range rdf:resource="&xsd:nonNegativeInteger"/>
</owl:DatatypeProperty>
...

```

Listing 3. Ontology.

```

...
<wsdlrdf:InterfaceOperation rdf:about="http://laris.eisti.fr#wsdl.
interfaceOperation(ServiceInterface/cropping)">
  <rdfschema:label>cropping</rdfschema:label>
  <sawSDL:modelReference rdf:resource="http://laris.eisti.fr/
ImageProcessing#cropping" />
...

```

Listing 4. From WSDL to RDF.

```

PREFIX xmlns:<http://www.w3.org/ns/wsdl-rdf#>
PREFIX sawSDL:<http://www.w3.org/ns/sawSDL#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX image:<http://laris.eisti.fr/ImageProcessing#>
SELECT DISTINCT ?name
WHERE {
  ?imageS sawSDL:modelReference image:SatelliteView .
  ?op wsdlrdf:interfaceMessageReference ?imageS.
  ?op rdfschema:label ?name
}

```

Listing 5. SPARQL queries.

### 2.8. Choreography contribution

The association of annotated services and SPARQL queries gives different use cases for dynamic composition. We can list them regarding variations onto domain or co-domain of the entire process:

- BPEL → WSDL<sup>n</sup>,  $n \in \mathbb{N}$ : searching services from BPEL;
- BPEL → BPEL: replacing a list of services by another one;
- WSDL<sup>n</sup> → BPEL,  $n \in \mathbb{N}$ : building BPEL

The workflow can be statically described as a BPEL file that chains different services. It is possible to prepare different files for different workflows, but it still remains static execution. With the previous listed cases, we can now dynamically change this execution. In the particular case of image processing services, the dynamic evolution of the process can improve the quality of the results. As an example, the very simple workflow that only make classification from *in situ* images give 75% of correct classes. The dynamic add of the cropping service in the workflow increases the ratio up to 90% because all the pictures are identically cropped.

### 3. Experimentation

In the previous section, we present a model that can improve the quality of an image processing workflow. This is done using a SOA architecture, where all services are semantically annotated. The services are implemented in Java and deployed on Apache Axis2. With the annotation and SPARQL queries, the workflow can be dynamically changed. We have implemented this model for a real case study of image processing for environmental analysis of glacier mass balance in Spitsbergen.

### 3.1. Workflow

Fig. 4 is an example of an image processing workflow. The images are the vertices and the tasks (web services) are the edges. State number 1 is the initial images. State number 6 is the result of the image processing which is the classification of the glacier phenomena. And all the paths correspond to the different ways of obtaining the classification.

The ideal path is {1,4,5,6}. The disturbances lengthen the path, by adding tasks to the processing sequence. The workflow has been defined in BPEL.

### 3.2. Ontology

The dynamic composition needs to query an ontology that contains the concepts and links of the domain. In the case of image processing services, we build an ontology with all the information about image, projection, classification and other transformations that appear in a classical workflow. Fig. 5 represents an interesting part of this ontology, we will use in the results part. It represents a complete hierarchy where every class in a subclass of the super-class “owl#Class” and every property is a subproperty of the super-property “owl#ObjectProperty”. Hence in the result part, we will be able to find a path between every concept that has been used to annotate our services.

### 3.3. Annotations

Once the ontology is complete, we use it to annotate the RDF mapping of web services. Each operation, input or output will be annotated with a concept from the ontology. Then, the different services will become implicitly linked altogether through the

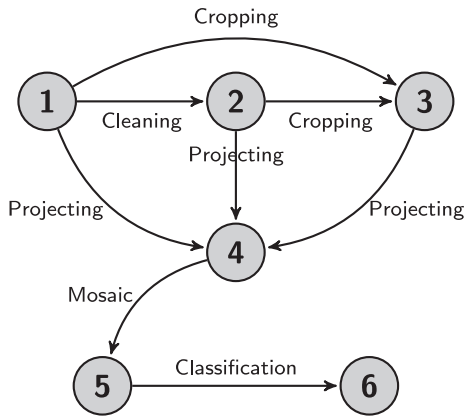


Fig. 4. Workflow of an image processing.

existing links between the concepts of the ontology. New notions, like services equivalence or services composition, appear with simple semantic inference and can be used to execute dynamic evolutions of the original workflow. Listing 6 represents a snippet of the web service projection after it has been annotated with SAWSDL markers.

### 3.4. Composition modification

We can replace one service by another that realizes an equivalent operation. There are many ways to execute cropping, depending, for example, on the polynomial degree of the transformation. The SPARQL request, in Listing 7, selects the name of all the services whose operation is cropping, as shown in Listing 8. With this result, we can discover and choose a new web service equivalent to the initial one in order to create a new workflow. The difference

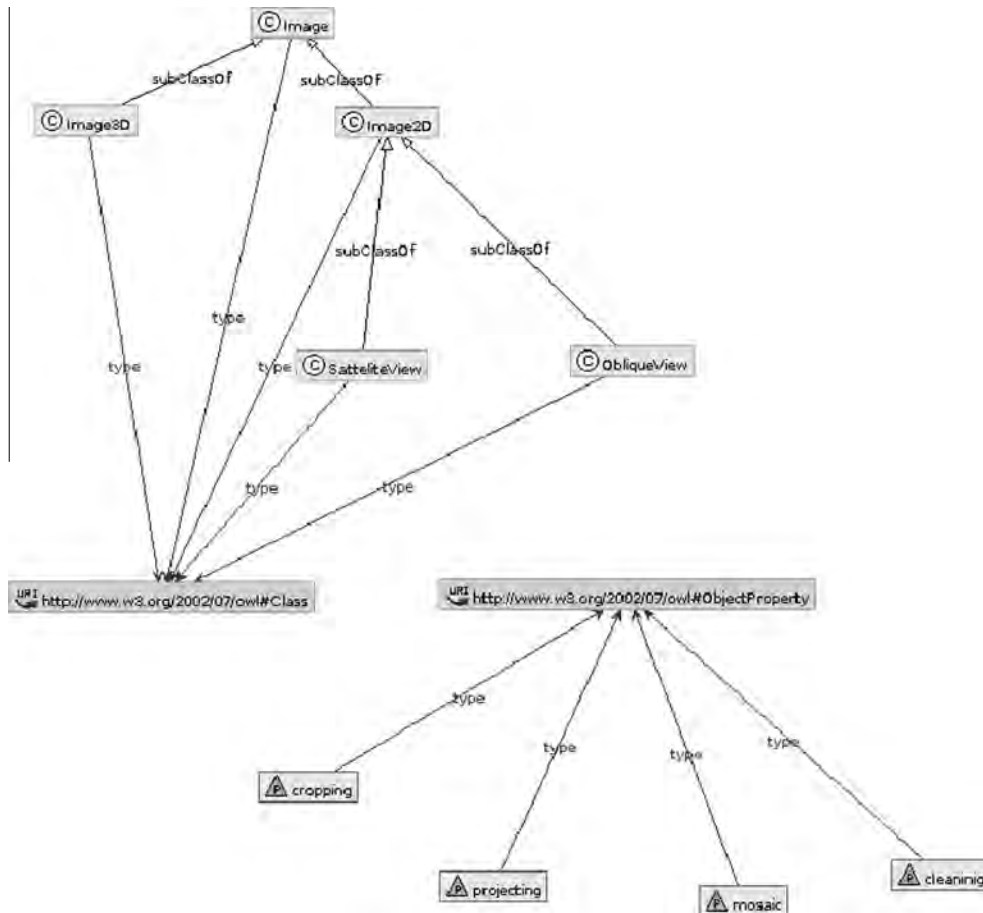


Fig. 5. Extraction of image processing ontology.

```

...
<wsdlrdf:InterfaceOperation rdf:about="http://laris.eisti.fr#wsdl.
  interfaceOperation(ServiceInterface/rubberSheeting)">
  <rdfschema:label>rubberSheeting</rdfschema:label>
  <sawSDL:modelReference rdf:resource="http://laris.eisti.fr/ImageProcessing
    #projecting" />
  ...
  <wsdlrdf:interfaceMessageReference>
  <wsdlrdf:interfaceMessageReference rdf:about="http://laris.eisti.fr#wsdl
    .interfaceMessageReference(ServiceInterface/rubberSheeting/Out)">
  <sawSDL:modelReference rdf:resource="http://laris.eisti.fr/
    ImageProcessing#SatelliteView" />
  ...

```

Listing 6. Annotated service.

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX sawsdl: <http://www.w3.org/ns/sawsdl#>
PREFIX wsdlrdf: <http://www.w3.org/ns/wsdlrdf#>
PREFIX image: <http://laris.eisti.fr/ImageProcessing#>
SELECT ?name
  WHERE {
    ?service rdf:type wsdlrdf:InterfaceOperation .
    ?service rdfs:label ?name .
    ?service sawsdl:modelReference image:cropping
  }
    
```

**Listing 7.** SPARQL request listing all the cropping services.

name
"cropping1order"
"cropping2order"
"cropping3order"

**Listing 8.** Result of the SPARQL query which selects the cropping services.



**Fig. 6.** Cropping services: superposition of the reference image and the result of the image cropping.

```

PREFIX ...
SELECT ?name ?name1 ?name2
WHERE {
  {
    ?service rdf:type wsdlrdf:InterfaceOperation .
    ?service rdfs:label ?name .
    ?service wsdlrdf:interfaceMessageReference ?interfaceIn .
    ?interfaceIn rdf:type wsdlrdf:InputMessage .
    ?interfaceIn sawsdl:modelReference image:ObliqueView .
    ?service wsdlrdf:interfaceMessageReference ?interfaceOut .
    ?interfaceOut rdf:type wsdlrdf:OutputMessage .
    ?interfaceOut sawsdl:modelReference image:ObliqueView
  } UNION {
    ?service1 rdf:type wsdlrdf:InterfaceOperation .
    ?service1 rdfs:label ?name1 .
    ?service1 wsdlrdf:interfaceMessageReference ?interfaceIn1 .
    ?interfaceIn1 rdf:type wsdlrdf:InputMessage .
    ?interfaceIn1 sawsdl:modelReference image:ObliqueView .
    ?service1 wsdlrdf:interfaceMessageReference ?interfaceOut1 .
    ?interfaceOut1 rdf:type wsdlrdf:OutputMessage .
    ?interfaceOut1 sawsdl:modelReference ?x .
    ?service2 rdf:type wsdlrdf:InterfaceOperation .
    ?service2 rdfs:label ?name2 .
    ?service2 wsdlrdf:interfaceMessageReference ?interfaceIn2 .
    ?interfaceIn2 rdf:type wsdlrdf:InputMessage .
    ?interfaceIn2 sawsdl:modelReference ?x .
    ?service2 wsdlrdf:interfaceMessageReference ?interfaceOut2 .
    ?interfaceOut2 rdf:type wsdlrdf:OutputMessage .
    ?interfaceOut2 sawsdl:modelReference image:ObliqueView .
    FILTER (!(?name1 = ?name2))
  }
  ...
}
    
```

**Listing 9.** SPARQL request listing all the services and composition of services which have oblique image as input and output.

name	name1	name2	...
"cropping"			...
"cleaning"	"cropping"	"cleaning"	...
...	"cleaning"	"cropping"	...

**Listing 10.** Result of SPARQL query which retrieves all the services and composition of services that has an oblique image as input and output.

between different cropping services is easily checked comparing the quality of resulting images in Fig. 6.

We can also replace a part of a composition service in the BPEL process by another equivalent part. In this case the new part has not systematically the same number of services as the initial part. We try to find a composition of services compatible with the input and output of initial BPEL part. In our case of image processing, the most simple composition is *projecting*, *mosaic* and *classification*. One possibility is to add services at the beginning of the workflow. The service (or composition of services) should have oblique image as input and output as we start with *in situ* pictures and because *projecting*'s input is an oblique image. The SPARQL request, in Listing 9, gives us all services and composition of services that have as an input and output an image representing an oblique view. In the case of a composition of services, the request filters out the compositions in which a service appears more than once because applying a service several times on the same picture will not give better results than applying it just once. The results, as shown in Listing 10, give many paths that modify an oblique image. We can lengthen the initial workflow with one (or a composition) of the services of the result, as represented in Fig. 4. For example: *cropping*, *cleaning*, *projecting*, *mosaic* and *classification*. The difference between the proposed services compositions is checked with the ratio of correct classes of the classification. These compositions form a set of BPEL processes that the end user can select.

#### 4. Conclusion

In this work, we have presented a complete model to add and use semantic annotations in a workflow of image processing services. The specificities of our database of photos and the need to modify static process very often were our major motivations.

With the new components we added in the image processing, we are now able to compute almost all the exploitable images from our database. With the dynamicity of the process, every kind of image is processed with a specific workflow, which does not have to be explicitly described. Moreover, with our solution, the addition of new available services will be automatically inserted in every workflow.

In order to reuse this model in a different context, an ontology must describe the new domain, then annotations and queries can be written instead of a very large list of specific processes. A static process that matches the classic workflow is used as a starting

point, and then it can be extended when new services are added or when the database changes.

#### Acknowledgement

We would like to thank the anonymous referees for their constructive remarks which have greatly improved this paper.

#### References

- Akkiraju, R., Sapkota, B., 2007. Semantic Annotations for WSDL and XML Schema – Usage Guide. W3C Working Group Note, World Wide Web Consortium (W3C). <<http://www.w3.org/TR/2007/NOTE-sawSDL-guide-20070828/>> (accessed 13.05.13).
- Bernard, E., 2011. Les dynamiques spatio-temporelles d'un petit hydrosystème arctique: approche nivo-glaciologique dans un contexte de changement climatique contemporain (bassin du glacier Austre Lovén, Spitsberg, 79°N). Ph.D. thesis. Université de Franche-Comté.
- Chinnici, R., Weerawarana, S., Moreau, J.J., Ryman, A., 2007. Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language. W3C Recommendation, World Wide Web Consortium (W3C). <<http://www.w3.org/TR/2007/REC-wsdl20-20070626/>> (accessed 13 May, 2013).
- Corripio, J.G., 2004. Snow surface albedo estimation using terrestrial photography. International Journal of Remote Sensing 25, 5705–5729.
- Hinkler, J., Petersen, S.B., Rasch, M., Hansen, B.U., 2002. Automatic snow cover monitoring at high temporal and spatial resolution, using images taken by a standard digital camera. International Journal of Remote Sensing 23, 4669–4682.
- Karande, A., Karande, M., Meshram, B., 2011. Choreography and orchestration using business process execution language for SOA with web services. International Journal of Computer Science Issues, 8.
- Kopecky, J., 2007. Web Services Description Language (WSDL) Version 2.0: RDF Mapping. W3C Working Group Note, World Wide Web Consortium (W3C). <<http://www.w3.org/TR/2007/NOTE-wsdl20-rdf-20070626/>> (accessed 13 May, 2013).
- Laffly, D., Bernard, E., Friedt, J.M., Martin, G., Tolle, F., Marlin, C., Griselin, M., 2012. High temporal resolution monitoring of snow cover using oblique view grounded-based pictures. Polar Record 48, 11–16.
- Newbery, K.B., Southwell, C., 2009. An automated camera system for remote monitoring in polar environments. Cold Regions Science and Technology 55, 47–51.
- Peltz, C., 2003. Web services orchestration and choreography. Computer 36, 46–52.
- Peterson, D., Gao, S.S., Biron, P.V., Malhotra, A., Thompson, H.S., Malhotra, A., Sperberg-McQueen, C.M., 2009. W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes. W3C Working Draft, World Wide Web Consortium (W3C). <<http://www.w3.org/TR/2009/WD-xmlschema11-2-20091203/>> (accessed 13 May, 2013).
- Prud'hommeaux, E., Seaborne, A., 2008. SPARQL Query Language for RDF. W3C Recommendation, World Wide Web Consortium (W3C). <<http://www.w3.org/TR/rdf-sparql-query/>> (accessed 13 May, 2013).
- Shapiro, R., 2002. A Comparison of XPD, BPML, and BPEL4WS. <<http://xml.coverpages.org/Shapiro-XPDL.pdf>> (accessed 13 May, 2013).
- White, S.A., 2004. Introduction to BPMN. Elements 15, 1–11.

### 9.3 Conclusion partielle

La sémantique nous a permis de diversifier la composition de services d'un BPEL de plusieurs manières. En se basant sur l'ontologie des services, les requêtes *SPARQL Protocol and RDF Query Language* (SPARQL) permettent de trouver des services différents au niveau de l'algorithme utilisé mais équivalents sur le traitement. Il est donc possible d'adapter un *workflow* de BPEL en remplaçant un service par un autre afin d'appliquer de nouvelles techniques de traitement. Également, les requêtes SPARQL permettent de modifier le nombre de services présents dans le *workflow* en cherchant et ajoutant des services qui respectent la continuité des types de données en entrée et sortie afin de garder un traitement cohérent entre les données.

À travers ces différentes modifications, nous pouvons adapter la composition de services à chacune des photos différemment perturbées. Cette adaptation engendre généralement une augmentation de services dans le *workflow*, rendant le traitement plus complexe mais générant également plus de données intermédiaires. Nous allons voir dans les chapitres suivants les avantages de l'architecture *cloud* pour le traitement d'une grande quantité de photos (chapitre 10), ainsi que des méthodes de réduction de dimensions (chapitre 11).



Troisième partie

Résultats





A man should look for what is, and not  
for what he thinks should be.

---

*Albert Einstein*



# Chapitre 10

## Comparaison des architectures

The only source of knowledge is  
experience.

---

Albert Einstein

Ce chapitre reprend le travail exposé lors du "13th International Circumpolar Remote Sensing Symposium" à Reykjavik, intitulé *Architecture comparison of an in-situ data processing application applied to glacier mass balance analyse* [Ranisavljević et al., 2014b].

### 10.1 Introduction : le projet Hydro-Sensor-FLOWS

Les fluctuations climatiques contemporaines ont un impact important sur les différents éléments de la cryosphère. Afin de comprendre la réactivité d'un glacier face à ces fluctuations, le projet HSF a analysé les flux liquides et solides du glacier du Lovén Est et suivi sa dynamique spatio-temporelle sur une période de 4 ans. Analyser les évolutions des paysages glaciaires et tout particulièrement analyser l'évolution de la couverture neigeuse d'un glacier requiert une acquisition fréquente et régulière de données sur une longue période. Puisque l'accès aux images satellite est onéreux et à cause des mauvaises conditions climatiques telles que d'épaisses couvertures nuageuses aux latitudes polaires, l'imagerie satellite quotidienne n'est pas toujours accessible ou adapté pour observer les évolutions temporelles rapides. La télédétection est complétée par un ensemble de caméras numériques automatisées et autonomes, placées tout autour du bassin versant du glacier, qui prennent trois photos par jour. Les perturbations météorologiques qui limitaient la vue de l'imagerie satellite se sont partiellement transformées en perturbations des photos *in situ* avec la présence de brouillard, de gouttes d'eau, de gel ou de neige sur l'objectif, gênant partiellement ou totalement la vue du paysage à analyser. Les photos sont récupérées 2 à 3 fois par an lors de missions sur le terrain, permettant non seulement la récupération des données, mais également l'entretien des appareils lorsque cela est nécessaire. Lors de ces maintenances, les appareils doivent être parfois enlevés de leur pied, n'assurant pas le repositionnement identique après réparation, perturbant le cadrage des photos. Outre ces différentes perturbations qui devront être traitées, la fréquence de 3 photos par jour et par appareil photo permet tout de même de générer une grande base de données. Afin de pouvoir extraire les informations nécessaires à l'analyse de l'évolution de la couverture

neigeuse du glacier du Lovén Est, les photos *in situ* doivent être traitées afin de permettre une reconstitution d'un équivalent de la vue satellite, puis d'identifier les différents éléments présents : glacier, neige, roche. L'extraction de ces informations nécessite plusieurs traitements : la projection des photos, la reconstitution de l'équivalent de la vue satellite par mosaïque des images projetées, puis la classification des différents éléments présents sur l'image reconstituée.

Grâce à l'utilisation d'appareils photo *in situ*, nous pouvons acquérir une grande base de données d'images (plus de 60 000). Cependant, afin d'obtenir l'analyse souhaitée et à cause des perturbations météorologiques, les traitements à appliquer sont complexes et lourds. De plus, chaque image n'est pas perturbée de la même manière. Ceci implique que les traitements ne seront pas les mêmes pour toutes les images.

Jusqu'à présent, les données étaient traitées à partir de plusieurs outils informatiques, étant plus ou moins automatiques en fonction des traitements et qui n'étaient généralement pas conçus pour traiter des données en masse. Avec de tels outils de traitements, seule une petite partie des données a été traitée. Depuis ces dernières années, l'informatique a fait de grands progrès dans le traitement de données massives, notamment avec l'arrivée du *cloud computing*.

Notre problématique est donc double : **Comment traiter facilement toutes les données avec toutes leurs spécificités ? Et quelle est l'avantage d'une architecture *cloud computing* ?**

## 10.2 Modèle et Méthode

Le *cloud computing* est un ensemble de ressources, de serveurs ou d'applications qui est offert en tant que "service" à un client (humain ou machine) et qui est accessible depuis Internet. Il assure un certain nombre d'avantages comme la scalabilité, la modularité, la facilité d'accès, etc. De plus, il supporte le stockage et le traitement de grandes quantités de données.

Il existe différentes structures de déploiement de *cloud computing* dont les 3 principales sont :

**Public** : Le client et le fournisseur de services font partie de deux entités différentes. Le client ne maîtrise pas la manière dont les données sont traitées ni le lieu où elles sont stockées. Le service auquel il a accès correspond simplement à une boîte noire qui lui permet d'appliquer différents traitements sur ses données.

**Privée** : Le client et le fournisseur de services appartiennent à la même entité (par exemple, une université). Le fonctionnement du service sous forme de boîte noire ne change pas, mais dans cette structure de déploiement, les machines allouées pour le traitement et le stockage des données sont connues.

**Hybride** : Cette configuration correspond à un mélange des deux configurations précédentes.

Face à la quantité de données à traiter, le *cloud computing* a la capacité de supporter une telle quantité. Par ailleurs, à cause des différentes perturbations, le traitement d'images devra varier. Afin de pouvoir rendre le traitement adaptatif, nous avons séparé celui-ci en plusieurs sous-tâches. En les implémentant de façon la plus "atomique" possible et sous forme de services, chacune de ces sous-tâches peut être assez générique afin de pouvoir s'adapter à d'autres situations que celle du traitement d'image pour l'analyse de la couverture neigeuse d'un glacier, et elles sont assez indépendantes pour pouvoir évoluer (par exemple, se mettre à jour) sans impacter toute la chaîne de traitement. La modularité du *cloud computing* rend cette évolution possible à la volée, c'est à dire

sans avoir à arrêter tout le système pour effectuer la mise à jour. De plus, les données n'étant pas toutes perturbées de la même manière, les traitements à appliquer seront plus ou moins sollicités, et l'ordre de ces différentes tâches pourra varier en fonction du type de perturbations subies. Ayant découpé le traitement d'image en un ensemble de sous-tâches, cette variation de traitement se manifesterà par une composition de services différente. Grâce à la scalabilité du *cloud computing*, la quantité de ressource nécessaire pour traiter l'ensemble des données pourra évoluer au moment du traitement en fonction du besoin.

Le schéma de la figure 10.1 récapitule le modèle de cette architecture. Chaque sous-tâche du traitement, correspondant à un service en particulier, est représentée par une boîte noire. Le déroulement du traitement d'image se fait en 5 étapes :

1. Charger l'ensemble des données à traiter sur le *cloud computing*.
2. Démarrer le traitement des données. Les traitements ne sont pas appelés individuellement. Le client fait appel à l'ordonnanceur qui :
  - (a) récupère depuis la BDD l'image à traiter ;
  - (b) en fonction des perturbations subies, l'ordonnanceur définit un enchaînement de services spécifiques et fait appel à chacun d'entre eux en fonction de l'ordre indiqué.
3. Une fois l'ensemble de la composition des services définis terminé, le résultat est renvoyé au client.

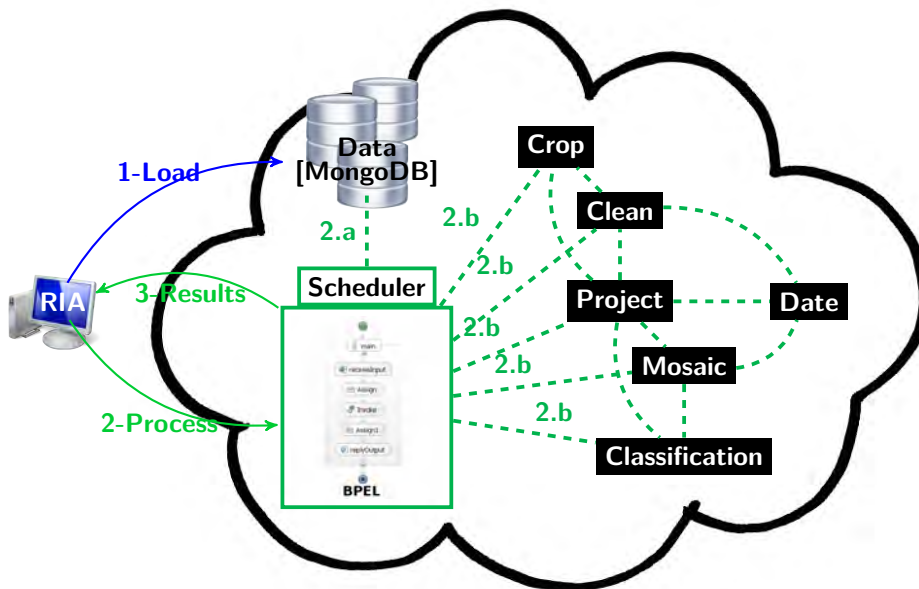


FIGURE 10.1 – Modèle d'une architecture de traitement d'images dans le *cloud computing*.

Dans le cadre de notre expérience, nous avons utilisé la BDD NoSQL MongoDB, adaptée pour le stockage de documents sur le *cloud*. Nous avons également utilisé un ordonnanceur BPEL couplé à la sémantique comme expliqué dans l'article [Ranisavljević et al., 2013] afin de le rendre plus

dynamique. Les services ont été implémentés en *Java*. Etant donné que certains des traitements nécessaires pour l'analyse du manteau neigeux d'un glacier à partir de photos *in situ* font encore partie du domaine de la recherche, nous les avons implémenté en tant que services "vides", c'est-à-dire n'appliquant aucune modification à l'image mais stockant uniquement le nouvel état de la donnée dans la BDD. Toutes les nouvelles informations générées par l'application d'un traitement (image, fichier de géolocalisation, ...) sont stockées dans la BDD. Nous avons fait le choix de conserver les états intermédiaires car ils peuvent être intéressants à analyser indépendamment du résultat final attendu. Donc nous avons fait le choix de les stocker afin de ne pas avoir à les re-générer.

Dans notre expérience, nous avons fait varier 3 paramètres :

**Les architectures :** un ordinateur personnel (4 CPU<sup>1</sup> @ 2.50 GHz, 8 GB RAM<sup>2</sup>), un serveur (8 CPU @ 2.00 GHz, 4 GB RAM) et un *cloud* privé composé de 40 machines (4 CPU @ 3.10 GHz, 2 GB RAM) non dédiées spécifiquement aux traitements d'images.

**Le nombre d'images à traiter :** en suivant presque une échelle logarithmique, nous avons choisi d'appliquer les traitements sur 10, 100, 1 000 et 5 000 photos afin de voir rapidement les différentes caractéristiques de chacune des architectures. Une photo acquise par les appareils photo fait en moyenne 4MB.

**La composition de services :** les lots d'images ont toujours été les mêmes, mais ces dernières ont été perturbées différemment, forçant la composition de service à varier.

Dans chacun des cas, pour toutes les images, nous avons mesuré, à partir du moment où le client faisait appel à l'ordonnanceur jusqu'à l'obtention des résultats, le temps écoulé (en secondes), la bande passante<sup>3</sup>, la charge du CPU ainsi que de la RAM. Ces différentes mesures ont été faites à travers le logiciel *Cacti* qui permet de mesurer les performances réseau et serveur.

### 10.3 Résultats et Discussion

Chacun des résultats présentés correspond à une moyenne effectuée sur 10 exécutions. Pour chacun des graphiques affichant les résultats, les architectures ont le code couleur suivant : ordinateur personnel en bleu, serveur en rouge, et *cloud* privé en jaune.

Par manque d'espace sur l'ordinateur personnel, il n'a pas été possible d'exécuter le traitement d'images pour 5 000 photos. En effet, étant donné que nous stockons chaque état intermédiaire, la BDD s'agrandit au fur et à mesure des traitements et dépasse la limite de l'ordinateur personnel utilisé pour l'expérience.

D'après le graphique des résultats en figure 10.2, nous remarquons deux points intéressants. Tout d'abord, avec peu d'images, l'utilisation du *cloud computing* n'est pas spécialement intéressant puisqu'il prend légèrement plus de temps qu'un ordinateur personnel ou un serveur. Ceci s'explique par une perte de temps du *cloud computing* à cause du transfert des informations entre les différents nœuds de calcul, de stockage et de contrôle, à l'opposé des deux autres architectures qui ont tout en local. Cependant, à partir de 1 000 photos, et c'est encore plus marqué pour 5 000 photos, le

---

1. Un *Central Processing Unit* (CPU) aussi appelé processeur, est un composant présent dans de nombreux dispositifs électroniques qui exécute les instructions machine des programmes informatiques

2. La *Random Access Memory* (RAM) aussi appelé mémoire vive, est la mémoire informatique dans laquelle sont rangées les informations (données, instructions, etc) en traitement par un appareil (par exemple, un ordinateur)

3. La bande passante indique la capacité d'un réseau à transmettre des informations

temps d'exécution pour le *cloud computing* est très inférieur à celui du serveur. L'utilisation des différents nœuds pour le *cloud computing* permet de distribuer les tâches sur plusieurs machines, permettant un gain de temps considérable malgré le temps de transaction entre les nœuds.

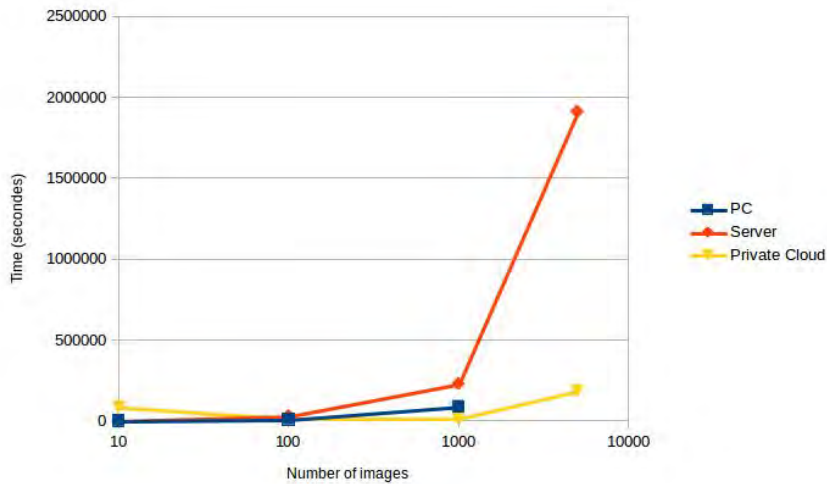


FIGURE 10.2 – Mesure du temps d'exécution pour les 3 architectures en fonction du nombre d'images à traiter.

L'utilisation de la bande passante est illustrée en figure 10.3. Étant donné que l'ordinateur personnel ou le serveur ne communique avec aucune autre machine pour effectuer le traitement, ces derniers n'utilisent pas la bande passante. À l'opposé, le *cloud computing* est basé sur la communication entre chacun des nœuds utilisés pour le traitement, donc l'utilisation de la bande passante est importante.

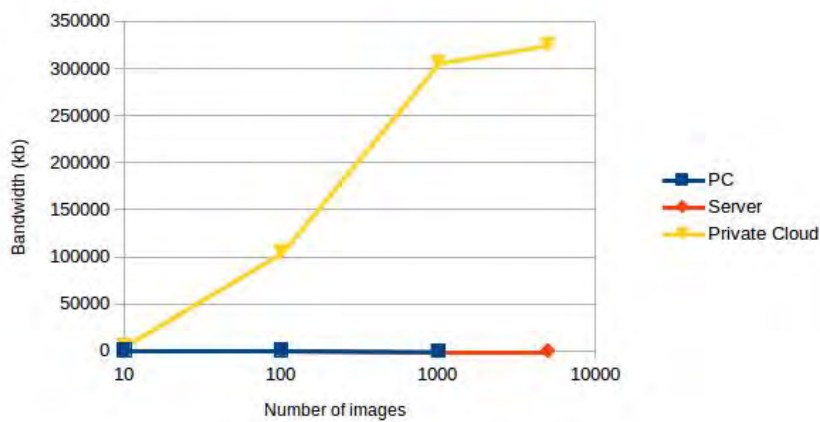


FIGURE 10.3 – Mesure de la bande passante utilisée pour les 3 architectures en fonction du nombre d'images à traiter.

La charge du processeur correspond à une moyenne des différents CPU disponibles dans chaque architecture, dont le calcul a été ramené à un pourcentage d'utilisation. Elle est renseignée en figure



10.4. Nous avons également mesuré l'utilisation de la RAM et illustré le pourcentage de mémoire libre en figure 10.5.

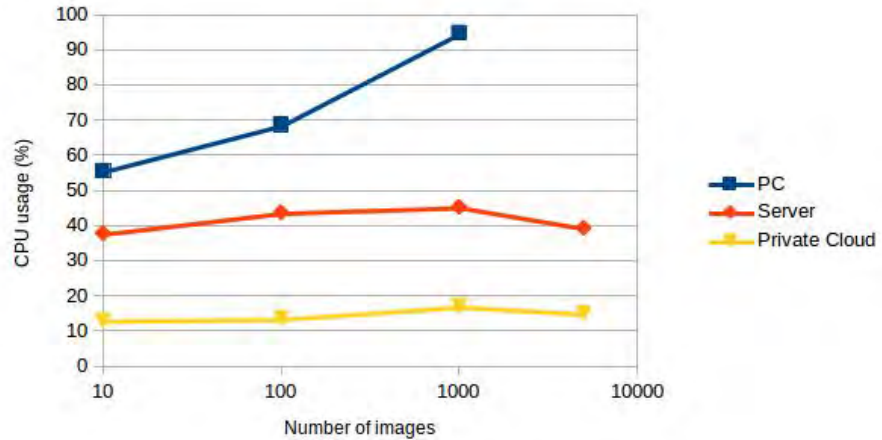


FIGURE 10.4 – Mesure de l'utilisation du CPU pour les 3 architectures en fonction du nombre d'images à traiter.

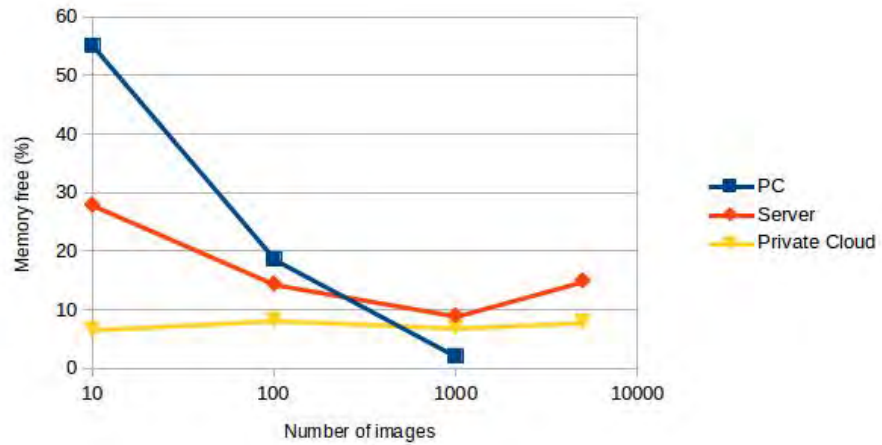


FIGURE 10.5 – Mesure de l'utilisation de la RAM pour les 3 architectures en fonction du nombre d'images à traiter.

Dans le cas de l'ordinateur personnel, nous remarquons que plus nous traitons d'images, plus les processeurs sont utilisés (en moyenne), et moins il y a de mémoire libre. Pour 1 000 photos, nous arrivons proche des limites de la machine autant au niveau du processeur que de la mémoire, ce qui explique également l'incapacité de la machine à traiter les 5 000 photos. Cette variation importante s'explique par l'utilisation constante des ressources, ainsi que par la difficulté à *Java* de gérer au mieux les ressources et notamment la libération des ressources de la RAM une fois ces dernières n'ayant plus d'utilité ou ayant été sauvegardées en BDD.

Dans le cas du serveur, étant donné que nous n'étions pas les seuls à l'utiliser, la mesure moyenne

du processeur est à peu près stable (environ 40-45%), et nous ne descendons pas non plus à une quantité d'espace mémoire libre trop faible.

Les mesures quasi constantes dans le cas du *cloud computing* sont plutôt faibles, autant pour l'utilisation du CPU que pour la mémoire libre. Elles s'expliquent également par la figure 10.3. Étant donné que les nœuds passent un certain temps à communiquer entre eux, les processeurs sont moins occupés à traiter les images mais plus à envoyer et recevoir les informations par paquets à travers le réseau. Les données étant la plupart du temps en transition, la RAM est utilisée pour effectuer les différentes transitions.

## 10.4 Conclusion

Le modèle que nous avons utilisé pour traiter nos images nous a permis de répondre à notre première problématique qui était : comment traiter facilement toutes les données avec leurs spécificités ? Traiter chacune des spécificités peut devenir laborieux si tous les cas sont à prendre en charge manuellement. En décomposant le traitement des images en plusieurs tâches, il n'est plus une seule tâche complexe mais devient la composition de plusieurs tâches moins complexes. Le nombre de services utilisés dans la composition ainsi que leur position dans l'enchaînement des tâches permet de s'adapter à chacune des spécificités. Grâce à la mise en place des traitements par service ainsi qu'à l'utilisation d'un ordonnanceur pour s'adapter aux différentes spécificités, le traitement n'est plus cantonné aux données pas ou peu perturbées mais peut s'appliquer à toutes les données. La limite actuelle est l'implémentation de services totalement autonomes pour les différentes tâches de traitement d'image. En effet, certaines tâches font toujours partie du domaine de la recherche (par exemple, définir l'utilisabilité d'une photo) et d'autres sont actuellement semi-autonome (le recadrage nécessite des données supplémentaires qui sont actuellement renseignées dans un fichier de configuration) et leur automatisation totale fait aussi partie du domaine de la recherche.

À travers nos différents tests, nous avons pu remarquer que pour traiter une grande quantité de données dans un temps raisonnable, le *cloud computing* répondait parfaitement à cela. Et même avec des machines non dédiées, nous obtenons des résultats intéressants. Dans notre expérience, nous nous sommes limités à 5 000 photos, ce qui correspond à un petit échantillon de photos à traiter comparé à la base de données actuelle, mais ceci nous a permis d'obtenir un ordre de grandeur principalement sur le gain de temps pour une architecture *cloud*.

Un *cloud computing* privé n'est pas quelque chose de répandu dans tous les milieux et cela nécessite un certain nombre de connaissances afin de le mettre en place et de le maintenir. De plus, l'architecture du *cloud* devient intéressante à partir d'une certaine quantité d'images à traiter. Il pourrait être intéressant de définir une mesure qui permettrait, en fonction d'un nombre de photos à traiter et supposant un nombre de services moyen pour la composition du traitement d'images, de définir quelle architecture serait la plus intéressante à utiliser, tout en intégrant la possibilité d'utiliser un *cloud* public, ainsi que son coût, en fonction du temps d'utilisation.



# Chapitre 11

## Réduction des dimensions des données

Pure mathematics is, in its way, the poetry of logical ideas.

---

*Albert Einstein*

Nous avons pu remarquer, dans le chapitre précédant, l'influence de la quantité de données sur le traitement et la limite qui est rapidement atteinte. À travers ce chapitre et une première ébauche de l'article à soumettre, nous nous penchons sur une étude statistique des données dans l'objectif de les réduire, afin de faciliter le transfert, le stockage, et peut-être même le traitement.

L'enjeu est donc de réduire la taille des données en s'efforçant de pouvoir extraire une bonne approximation des quantités d'intérêt et d'informations pour effectuer la tâche désirée en aval.

### 11.1 Une masse de données géographiques grandissante

Le réchauffement climatique contemporain a des conséquences visibles depuis plusieurs dizaines d'années en arctique par la fonte accélérée des calottes glaciaires et des glaciers de piémonts. De nombreuses études annoncent la disparition de la banquise et des glaciers arctiques d'ici une trentaine d'années alors que d'autres, pour peu que leur gls :bassinVersant soit suffisamment haut en altitude, trouveront un nouvel équilibre [Salas et al., 2010, Kergomard, 2007, Vaughan et al., 2013]. Dans tous les cas, de vastes espaces morainiques auront été libérés des glaces et les surfaces autrefois protégées commenceront à fondre en même temps que la végétation s'y installera [Kaplan and New, 2006].

Dans ces conditions, parallèlement aux études dressant des bilans globaux à l'échelle de l'arctique, il convient de s'intéresser aux rythmes de la fonte des glaciers et des bilans hydrologiques des bassins versants concernés. L'eau qui s'écoule en aval est composée en partie des précipitations actuelles mais aussi de la fonte du pergélisol, de la neige et de la glace, dans des proportions variables. Il est important de pouvoir quantifier, qualifier et dater toutes les eaux d'écoulements qui dressent le bilan hydrologique dans l'objectif de promouvoir de futures modélisations prospectives qui anticiperont sur la réalité des espaces déglacés de demain. Le projet glshsf a focalisé cette recherche sur

un site expérimental au Spitsberg, le glsassinVersant du glacier Lovén est.

En aval, voire en plusieurs points stratégiques des drains, les écoulements peuvent être facilement quantifiés avec des stations automatiques. Des prélèvements sont également réalisés à intervalle de temps régulier pour être ensuite chimiquement et physiquement analysés. Les précipitations sont enregistrées par des stations météorologiques distribuées selon un transect altitudinal qui parcourt le grand axe de l'appareil glaciaire. L'état de la surface du glacier et le suivi de la fonte neigeuse peuvent, en théorie, être cartographiés par l'utilisation de données satellitaires optiques mais les conditions climatiques limitent considérablement les fréquences d'acquisition. Les coûts des images étant prohibitifs, la solution retenue fut celle de données Formosat tous les 15 à 30 jours. De nos jours, le satellite Sentinel 2, mis en orbite en juin 2015, donne accès gratuitement à des données acquises tous les cinq jours, ce qui change radicalement la donne. Pour offrir une solution de suivi à haute fréquence temporelle, un système d'acquisition *in situ* a été développé. Six caméras autonomes programmées pour acquérir trois images par jour ont été positionnées sur les hauteurs du bassin versant pour couvrir visuellement toute la surface de celui-ci. Elles fonctionnent depuis mars 2007 et la base de données ainsi constituée compte aujourd'hui plus de 40 000 images qu'il convient de traiter pour les mettre en projection géographique, les assembler en mosaïque et les catégoriser pour distinguer la neige, la glace et le minéral afin d'estimer la lame d'eau de fonte par un modèle de type fusion-jour.

L'acquisition de ces images *in situ* génère :

- beaucoup de photos à récupérer, à stocker, à transférer et à traiter ;
- des images très corrélées puisque les caméras sont fixes et la fréquence d'acquisition est élevée.

## 11.2 Réduction de dimension

On considère un échantillon constitué de  $l$  images  $X_i$ ,  $i = 1, \dots, l$  et chaque image est considérée comme une matrice  $4224 \times 2376$  pixels. Dans cette partie, la réduction des dimensions des données passe par deux étapes :

1. Etape 1 : la réduction de la taille des photos à traiter à l'aide de la transformée de Haar.
2. Etape 2 : la réduction de la taille de l'ensemble des données obtenues après la transformation de Haar.

### 11.2.1 La réduction de la taille d'une image

On considère une image  $X_i$  sous la forme d'une matrice  $M \times N$ . On souhaite représenter et traiter  $X_i$  en utilisant un volume minimum de données :

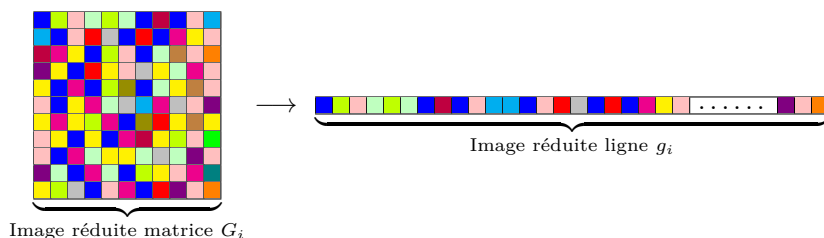
$$X_i = \sum_{k=1}^s \alpha_k \Psi_k.$$

Pour trouver la meilleure représentation parcimonieuse de  $X_i$  (peu de composantes sont non-nulles), on a fait un choix orienté selon un critère de connaissances à **priori** : les paquets d'ondelettes de Haar. La transformée en ondelettes est une technique de compression à base d'ondelettes qui offre une plus grande finesse au niveau de l'analyse du signal :

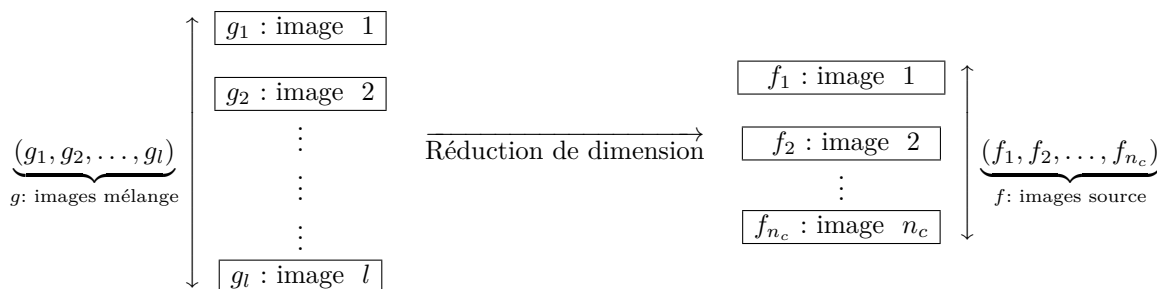
- elle permet de mieux s'adapter aux propriétés locales de l'image,
- elle présente l'avantage de faire ressortir à la fois les grandes variations et les détails de l'objet analysé.

## 11.2.2 La réduction de dimension des données réduites

Une image  $X_i$  est décrite par  $4224 \times 2376$  pixels que l'on réduit par la transformée de Haar. Cette dernière transformée, notée  $G_i$  est donc donnée sous forme d'une matrice  $528 \times 297$  que l'on a réécrite sous forme d'un vecteur ligne.



Chaque image utilisée dans ce travail sera donc décrite par  $n_p$  observations. Une observation est donc définie dans l'espace  $\mathbb{R}^l$  et chaque image issue d'une caméra parmi les 6 caméras est considérée comme une variable quantitative. Une réduction de dimension consiste à obtenir une représentation des  $n_p$  observations dans un nouvel espace  $\mathbb{R}^{n_c}$ , avec  $n_c \ll l$ .



Cette diminution du nombre de variables décrivant les données aura pour objectif de :

1. réduire la taille de données à traiter, tout en conservant au mieux "l'information utile" ;
2. améliorer le rapport signal / bruit et supprimer des variables non pertinentes. La prise des images du glacier est souvent redondante donc on peut définir des images comme étant des variables non pertinentes ;
3. améliorer la " lisibilité " des données en mettant en évidence des relations entre variables (images) ou groupes de variables (images issues de la même caméra) ou en facilitant la visualisation. Cette lisibilité devrait nous permettre de distinguer la neige de la glace.

## 11.2.3 Position du problème

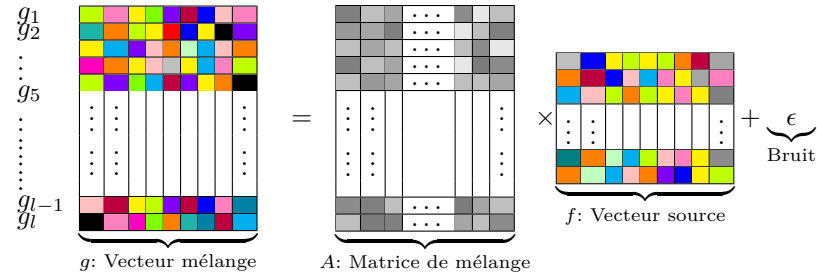
L'objectif de la réduction des dimensions des données sur les photos serait de diminuer le nombre d'images afin de ne stocker et de ne transférer qu'un nombre réduit d'images, mais tout en étant capable de reconstituer l'ensemble des images pour le traitement.

**Formulation de la problématique :** soient  $\mathbf{f}$  et  $\mathbf{g}$  deux vecteurs d'images appelés respectivement source et mélange (ce dernier correspondant aux images *in situ*),  $\mathbf{A}$  une matrice appelée

matrice de mélange et  $\epsilon$  un vecteur d'erreur. Peut-on trouver une décomposition de la forme

$$\mathbf{g} = \mathbf{A}\mathbf{f} + \epsilon \quad (11.1)$$

qui nous permettrait de diminuer le nombre d'images d'origine ?



**Éléments de réponse :** l'équation 11.1 fait partie de la catégorie des problèmes dit très mal posés car, connaissant  $\mathbf{g}$ , trouver les valeurs de  $\mathbf{A}$  et  $\mathbf{f}$  donne une infinité de solutions. Afin de trouver une solution unique, il faut ajouter un certain nombre de contraintes au problème initial.

L'approche de méthodes de réduction de dimension utilisée dans ce travail est fondée sur la transformation de variables. Cette transformation consiste à construire de nouvelles variables à partir des variables initiales. L'axe principal du travail est basé sur l'analyse factorielle linéaire qui est à caractère exploratoire, adaptée à des données décrites par des variables quantitatives. On exploitera le caractère de corrélation des images d'une même caméra, le caractère indépendant des images issues de différentes caméras, la distribution des variables, etc

Dans les deux cas, l'objectif est de toujours être capable de recréer la photo d'origine, avec le moins d'erreur possible, afin de pouvoir appliquer les différents traitements pour la suite de l'analyse sur les données.

### 11.3 Article draft : *"Wavelet Transform and Dimensionality Reduction methods for analysing in-situ sensing images database"*

# Wavelet Transform and Dimensionality Reduction methods for analysing *in-situ* sensing images database

Elisabeth RANISAVLJEVIĆ<sup>a</sup>, Ali MOHAMMAD-DJAFARI<sup>b,\*</sup>, Nisrine FORTIN CAMDAVANT<sup>a</sup>, Astrid JOURDAN<sup>a</sup>, Yannick LE NIR<sup>a</sup>, Florent DEVIN<sup>a</sup>, Dominique LAFFLY<sup>c</sup>

<sup>a</sup>*EISTI, 2 boulevard Lucien Favre, CS 77563, 64 075 PAU CEDEX, France*

<sup>b</sup>*Laboratoire des Signaux et Systèmes, UMR 8506 CNRS-SUPELEC-UNIV PARIS SUD, SUPELEC, Plateau de Moulon, 3 rue Joliot-Curie, 91192 Gif-sur-Yvette, France*

<sup>c</sup>*University of Toulouse, GEODE UMR 5602 CNRS, Maison de la recherche, 5 allées Antonio Machado, 31 058 TOULOUSE CEDEX, France*

---

## Abstract

In this paper, we present some preliminary results of a few image processing steps in size reduction of an image data set, such as Haar wavelet transform and Dimensionality Reduction. The data set is constituted of *in-situ* images obtained from the French Scientific missions in Spitzberg glacier.

*Keywords:* Haar Wavelet Transform, Dimensionality Reduction, Principal Component Analysis (PCA), Independent Component Analysis (ICA), Spitzberg glacier, *In-situ* imaging

---

## 1. Introduction

The contemporary global warming had has visible consequences for several decades in the Arctic by the accelerated melting of ice caps and glaciers piedmonts. Whatever the scenario adopted by Humanity to fight against global warming, many studies announce the disappearance of Arctic sea ice and glaciers from now to thirty years while others, as long as their catchment basin is sufficiently high altitude, will find a new equilibrium. In all cases, extensive moraine areas would have been ice-free and formerly protected surfaces will start to melt as the same time as vegetation will settle.

---

\*Corresponding author



Therefore, as for the circumpolar tundra, large quantities of organic carbon dioxide from the soil or SOC (Soil Organic Carbon) will be exchanged with the atmosphere thus contributing to further increase the greenhouse effect.

Under these conditions, in parallel with studies evaluating overall balances on the scale of the Arctic, it is worth considering the rhythms of melting glaciers and water balances of catchment basins concerned. The water that flows downstream consists partly of current rainfall but also to melting permafrost, snow and ice in varying proportions. It is fundamental to quantify, qualify and date all waters flow that take part in water balance in order to promote future prospective models that anticipate the reality of tomorrow deglaciated spaces. The *Hydro Sensor Flow* program (M. Griselin, C. Marlin et D. Laffly ; ANR and IPYprogram – *International Polar Year 2007-2008*) focused its research on an experimental site in Spitsbergen, the catchment bassin of East Lovén glacier.

Downstream or even on several strategic points of the drains, flows can be easily quantified with automatic stations. Samples are also made at regular time interval to then be physically and chemically analyzed. The rainfall are recorded by weather stations which are distributed along an altitudinal transect that runs through the long axis of the glacier. The state of the surface of the glacier and monitoring of snow melt can theoretically be mapped by the use of optical satellite data but weather conditions considerably limit the acquisition frequencies. The cost of the images was prohibitive, the solution adopted was FORMOSAT data every 15 days. Today the Sentinel 2 satellite put into orbit in June 2015 provides free access to data acquired every five days and radically changes the situation. To provide high temporal frequency tracking solution, *in-situ* data acquisition system has been developed, six autonomous cameras programmed to acquire three images per day have been positioned on the heights of the catchment bassin to visually cover the entire surface of the latter. They have worked since March 2007 and the database thus constituted has more than 40 000 images which should be processed to put them in geographical projection, assemble them on mosaic and categorize them to distinguish snow, ice and to estimate the mineral meltwater blade model type merger-date. All these treatments have a processing cost and therefore require to use size reduction methods to be fully operational.

## 2. Size reduction

Reducing the dimensions of an image data set can be done in two ways: reducing the size of the images and reducing the number of images in the data set.

About reducing the size of the images, several methods have already proven themselves, such as Discrete Fourier Transform, Discrete Sine or Cosine Transform, Discrete Wavelet Transform (Haar, Daubechies, ...) ... [Porwik and Lisowska (2004); Pratt (2002); Wang (2001)]. These are orthogonal functions and with only additions and subtraction, their forward and inverse transforms can be easily implemented. We chose the Haar wavelet transformation as it is the main method used for image compression in geomatic [Liu et al. (2005)].

Applying an Haar wavelet transformation on a 2D-image of size  $m \times n$  generate 4 new images of size  $\frac{m}{2} \times \frac{n}{2}$  [Talukder and Harada (2010)] (example figure 1): the top-left image is a equivalent of the initial image and the top-right, bottom-left and bottom-right are respectively the residual information of the vertical, horizontal and diagonal details of the image.

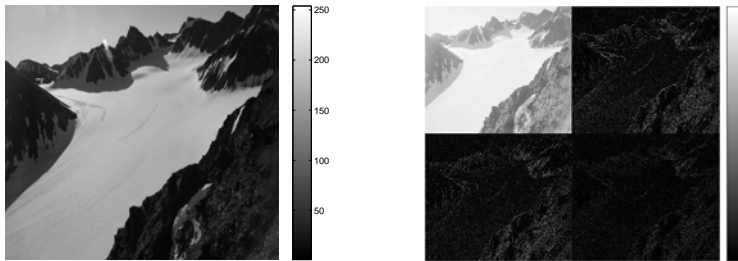


Figure 1: Haar WT: left the initial image, right the 4 images generated (logarithmic scale)

This transformation can be applied again at the  $i$ -level image, and so on (figure 2), as long as the residue is negligible. The low resolution image obtained at the end is still very similar to the high resolution image (figure 3).

As for reducing the quantity of images, there are also several methods, in factor analysis [Shlens (2014); Abdi and Williams (2010); Cardoso (2002)] as in inverse problem [Park and Casella (2008); Tibshirani (1996); Davis et al. (1997); Aharon et al. (2006)] , and they all have approximation error in the inverse transform.

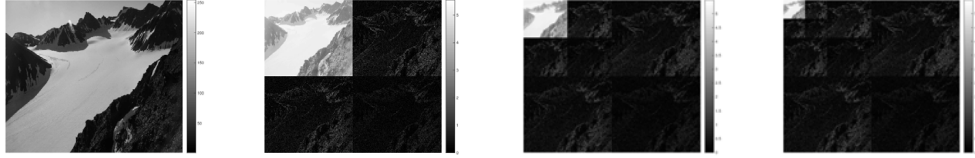
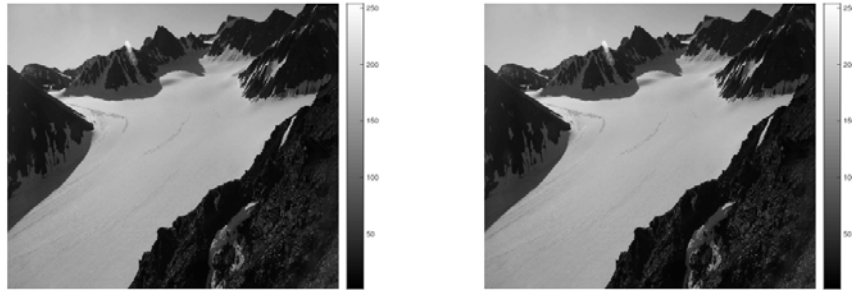


Figure 2: Successive Haar WT (logarithmic scale)



(a) HR image

(b) LR image

Figure 3: Using WT to reduce the size of images: a) HR image, b) LR image obtained by Haar Wavelet transform using a 3-level WT and keeping only the LR image.

In this preliminary study, we will focus only on methods of factor analysis, and we will compare some of them in order to determine which method regenerates the *in-situ* images the closest to their original form.

### 3. Factor analysis

Knowing an image vector  $\mathbf{g} = (\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_l)$ , with  $l$  the number of images, our goal is to compress the size of the data set in order to keep only the important informations. For that, we find a source image vector  $\mathbf{f}$  of size  $n_c$  smaller than  $l$ , called *principal components*, which are a linear combination of the variable  $\mathbf{g}$  :

$$\mathbf{g} = \mathbf{A}\mathbf{f} + \epsilon \quad (1)$$

with  $\mathbf{A}$  a mixing matrix of size  $l \times n_c$  and  $\epsilon$  an approximation vector error of size  $l$ .

The equation (1) is an ill-posed problem as there is an infinity of solutions. In order to find a single solution, we must add a number of constraints to the original problem.

Suppose there is no approximation error  $\epsilon$  and  $\mathbf{A}$  is invertible. The problem (1) is written:

$$\mathbf{g} = \mathbf{A}\mathbf{f} \quad (2)$$

Let  $\mathbf{B} = \mathbf{A}^{-1}$ . The source vector is then :

$$\hat{\mathbf{f}} = \mathbf{B}\mathbf{g} \quad (3)$$

The matrix  $\mathbf{B}$  is selected such that the components of  $\hat{\mathbf{f}} = (\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_{n_c})$  are:

1. the least **correlated** as possible: it is the Principal Component Analysis (PCA) method and its variants ;
2. the most **independent** as possible: it is the Independent Component Analysis (ICA) method.

### 3.1. PCA Method

The principal components analysis (PCA) is a descriptive method not based on a probabilist model. It is a size reduction tool for removing the redundancy in a set of correlated variables [Abdi and Williams (2010); Shlens (2014)]. For the *in-situ* images, we seek which images are the most correlated between them. The variance-covariance matrix measure the connexion between the different descriptors.

The image vector  $\mathbf{g} = (\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_l)$  is organised to have a size matrix of  $l \times n_p$ , with  $n_p$  the number of pixel in an image. Under the hypothesis that the mean of  $\mathbf{g}$  is null :

1. We calculate the covariance matrix of  $\mathbf{g}$ :

$$\mathbf{C}_{\mathbf{g}} = cov(\mathbf{g}) = \frac{1}{n_p} \mathbf{g}'\mathbf{g} \quad (4)$$

2. We apply a singular value decomposition (SVD) of  $\mathbf{C}_{\mathbf{g}}$ :

$$\mathbf{C}_{\mathbf{g}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^t \quad (5)$$

with  $\mathbf{\Sigma}$  the matrix of  $\mathbf{C}_{\mathbf{g}}$ 's eigenvalues sorted in descending ordre,  $\mathbf{U}$  the orthogonal matrix of  $\mathbf{C}_{\mathbf{g}}$ 's eigenvectors and  $\mathbf{V} = \mathbf{U}$ .

3.  $\mathbf{g} = \mathbf{A}\mathbf{f} \Rightarrow \mathbf{C}_g = \mathbf{A}\mathbf{C}_f\mathbf{A}' = \mathbf{U}\Sigma\mathbf{U}^t$  where  $\mathbf{C}_f$  is the covariance matrix of  $\mathbf{f}$ .

We choose  $\mathbf{A} = \mathbf{U}$ . In order to reduce the size of the data set, we only keep the first  $n_c$  columns of  $\mathbf{U}$ . Choosing the value of  $n_c$  is an open problem [Abdi and Williams (2010)], but there different ways of doing it:

- visually define the bending point, e.g. when the curve representing the eigenvalues goes from “steep” to “flat” (image 4);

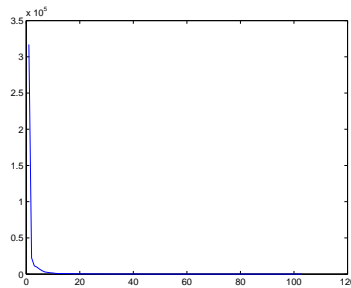


Figure 4: Eigenvalues of the covariance matrix

- keep the eigenvalues which are greater than the average of the eigenvalues;
- keep the eigenvalues which are greater than 1;

- use the percentage of total inertia explained as  $I_e = \frac{\sum_{k=1}^{n_c} \lambda_k}{\sum_{k=1}^L \lambda_k}$  and

keep  $n_c$  according to the quantity of information wanted.

4. We deduce  $\hat{\mathbf{f}} = \mathbf{A}'\mathbf{g}$

An example of the principal components is given in figure 5.

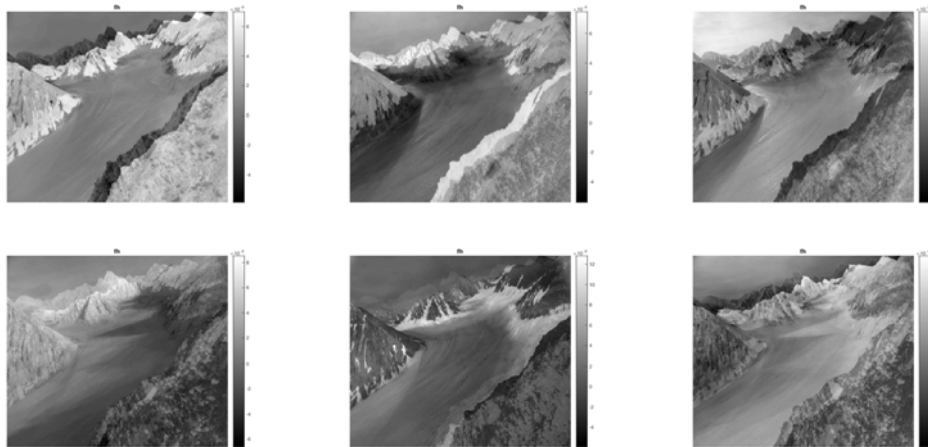


Figure 5: 6 first principal components of PCA

### 3.2. PPCA Method

The absence of a generative data model and an associated probability density is noticed in PCA. The probabilistic principal components analysis (PPCA) is derived from a mixing model with the hypotheses of an isotropic noise and an *a priori* gaussian [Tipping and Bishop (1999)]. The generative problem is formulated according to the following expression:

$$\mathbf{g} = \mathbf{A}\mathbf{f} + \mu + \epsilon \quad (6)$$

with  $\mu$  as the data mean and  $\epsilon$  a gaussian noise  $\epsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$ . The likelihood is then given by:

$$P(\mathbf{g}|\mathbf{f}) = \mathcal{N}(\mathbf{g}|\mathbf{A}\mathbf{f} + \mu, \sigma^2 \mathbf{I}) \quad (7)$$

By maximum likelihood, the analytical solution is obtained:

$$\hat{\mathbf{A}} = \mathbf{U}(\mathbf{\Lambda} - \sigma^2 \mathbf{I})^{1/2} \mathbf{R} \quad (8)$$

with  $\mathbf{R}$  an arbitrary orthogonal rotation matrix,  $\mathbf{\Lambda}$  the diagonal matrix of the first  $n_c$  eigenvalues of the covariance matrix of  $\mathbf{g}$  sorted in descending order, and  $\mathbf{U}$  the matrix of the corresponding first  $n_c$  eigenvectors.

An estimator of the vector  $\hat{\mathbf{f}}$  is the mean of the *a posteriori* distribution given by:

$$\hat{\mathbf{f}} = (\sigma^2 \mathbf{I} + \hat{\mathbf{A}}' \hat{\mathbf{A}})^{-1} \hat{\mathbf{A}}' (\mathbf{g} - \mu) \quad (9)$$

### 3.3. FACTORAN Method

The factor analysis (FACTORAN) [Jöreskog (1967)] corresponds to a generative model similar to PPCA with the difference assumed that the gaussian noise added is diagonal:  $\epsilon \sim \mathcal{N}(0, \Psi)$  with  $\Psi = \mathbf{diag}(\psi_1, \dots, \psi_1)$ .

### 3.4. PPCA-MV Method

In probabilistic principal components analysis with missing values (PPCA-MV)[Verbeek (2009)], we use the *Expectation-Maximization (EM)* algorithm in order to address the problem of missing value based on the *maximum-likelihood estimation* of the data set.

The *EM* algorithm proceeds by considering the observed data  $\mathbf{g}$  incomplete. It increases the observations with hidden variables  $\mathbf{z}$ .

*EM* is an iterative algorithm that operates in two separate steps:

1. The expectation phase *E* is to estimate the unknown data by estimating the likelihood of the complete data:

$$\mathbb{E}_{\mathbf{z}|\mathbf{g},\theta}(\log(P(\mathbf{g}, \mathbf{z}|\theta))) \quad (10)$$

with  $\theta$  the density parameters of  $\mathbf{g}$  and  $P(\mathbf{g}, \mathbf{z}|\theta) = P(\mathbf{z}|\mathbf{g}, \theta)P(\mathbf{g}|\theta)$ .

2. The maximization phase *M* is to maximize the estimated likelihood to determine the new value of the parameter  $\theta$ . At iteration  $n + 1$ ,  $\theta$  is determined by:

$$\theta_{n+1} = \operatorname{argmax}_{\theta}(\mathbb{E}(\log(P(\mathbf{g}, \mathbf{z}|\theta)))) \quad (11)$$

### 3.5. ICA Method

PCA seeks the non-correlated and orthogonal components to best represent the data. The independent components analysis (ICA) seeks the components the most statistically independent to each other [Cardoso (2002)]. They will be therefore not correlated but not necessarily orthogonal.

In ICA, the main idea is to build a good estimation of the source vector  $\mathbf{f}$  and of the linear transformation  $\mathbf{A}$ . We build an auxiliary vector

$$\mathbf{y} = \mathbf{E}\mathbf{g} \quad (12)$$

with  $\mathbf{E}$  the separation matrix.

1. We seek  $\mathbf{y}$  with the maximum of independent components. To do this, an estimate of the separation matrix is determined by maximizing a measure of independence of the components of  $\mathbf{y}$ . We choose the measure of independence of Kullback-Leiber  $KL(.,.)$  and we get :

$$\hat{E} = \underset{E}{\operatorname{argmax}} \left( KL(p(\mathbf{y}), \prod_i p(\mathbf{y}_i)) \right) \quad (13)$$

2.  $\mathbf{g} = \mathbf{A}\mathbf{f} \Rightarrow \hat{\mathbf{y}} = \hat{\mathbf{E}}\mathbf{g} = \hat{\mathbf{E}}\mathbf{A}\mathbf{f}$ . If we impose the constraint  $\mathbf{A}$  is reversible and that  $\mathbf{A}^{-1} = \hat{\mathbf{E}}$  then  $\hat{\mathbf{y}} = \mathbf{f}$ .  $\hat{\mathbf{y}}$  is considered to be an estimator  $\mathbf{f}$ . We deduce  $\hat{\mathbf{f}} = \hat{\mathbf{E}}\mathbf{g}$ .

An example of the principal components is given in figure 6.

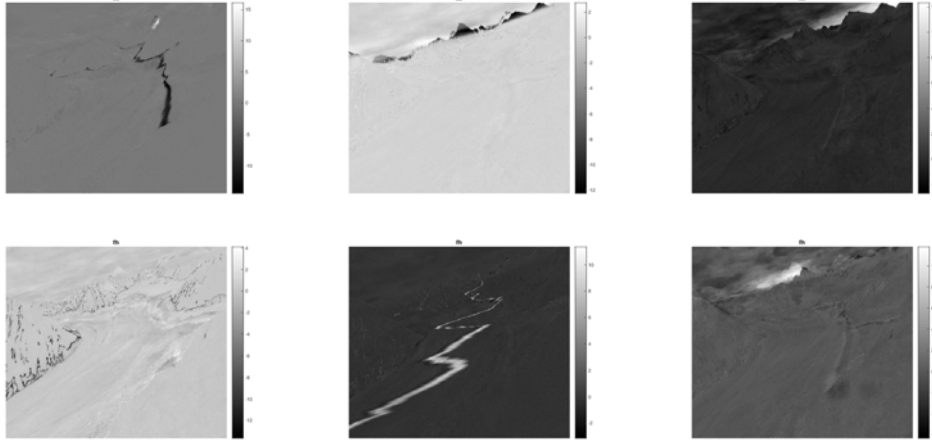


Figure 6: 6 first principal components of ICA

#### 4. Discussion : quality of reconstruction

We have six cameras positioned on the heights of the catchment basin. The results are PSNR (Peak-Signal-to-Noise Ratio) measures calculated as follow [Hore and Ziou (2010)]:

$$PSNR(\mathbf{f}, \mathbf{g}) = 10. \log_{10}(255^2 / MSE(\mathbf{f}, \mathbf{g})) \quad (14)$$

$$MSE(\mathbf{f}, \mathbf{g}) = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (\mathbf{f}_{ij} - \mathbf{g}_{ij})^2 \quad (15)$$



PSNR allows to numerically quantify the quality of the reconstruction of an image. This measure allows to quantify the distortion between two images.

These images have been reconstructed after 5 different processes : FACTORAN, PCA, PPCA, PPCA-MV and ICA. In order to keep at least 98% of the total inertia, we selected  $n_c = 20$  principal components. This study is mainly based on three goals :

1. An exploratory and descriptive study of the measurements obtained on the camera 1 with the 5 different processes. The sample consists of 103 observations.
  - The graphical representation (Tukey boxplot) will highlight one or more central tendency parameter and a dispersion parameter. This exploration will allow us to visually grasp the magnitude of the PSNR measure and its dispersion and identify outliers. This approach will lead us to analyze the source images (reduced in size by the Haar wavelet transform) with those reconstructed and having the same reduced size.
  - The graphical representation (histogram) will propose a probabilistic model for the distribution of the measurement for each process. This will allow us to choose a test of process comparison: parametric or nonparametric test.
2. A descriptive study of all the measures obtained from the 6 cameras after each process. We have 5 samples (5 processes) and each sample is made up of PSNR measurements of the 6 cameras thus 967 measures.
  - The shape of the histogram of the frequencies allow us to choose the test to be performed between processes.
  - The variance analysis covers a set of technical tests and estimation for assessing the effect of the process on the PSNR measurement and returns to the simple case of comparing means of several Gaussian samples [Saporta (1990)]. Coming from real experiment field, it is relevant for the digital experimentation since it identifies the main features of the behavior of a model [Faivre et al. (2013)].
  - The analysis of variance ANOVA allows us to validate or invalidate the hypothesis of substantive equality of process on the images.

3. A sample comparison test to determine the suitability of a process from another. We do a *Student* test which is a statistical test to compare the means of two groups of samples. It is therefore whether the averages of the two processes are significantly different in statistical terms.
- 4.1. *Results : exploratory and descriptive study of the measurements obtained on the camera 1*

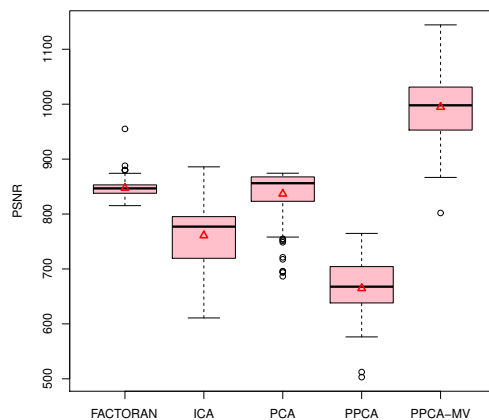


Figure 7: Boxplot of PSNR measures for the 5 processes on the camera 1 ( $\Delta$  represents the mean value, and  $\circ$  represents the outlier values).

We can see a very significant difference between the 5 processes (figure 7). Outliers are distinguished in FACTORAN and PCA processes. Table 1 summarizes the different images that provide outliers measures.

Table 1: Images with outlier value of PSNR (■above and ■below)

Image number	4	5	9	10	11	13	14	15	16	17	20	37	75	81	101	103
Process FACTORAN			■	■				■			■	■				
Process ICA																
Process PCA	■	■			■	■	■	■	■	■					■	■
Process PPCA																
Process PPCA-MV													■			

The boxplots (figure 7) show a stable variability results from FACTORAN process as the interquartile range is very narrow. We notice an asymmetrical distribution for ICA and PCA processes, the measure’s mean being lower than the median.

It will highlight the quite high PSNR values of the PPCA-MV process and that more than 75% of its values are larger than the measure’s mean of FACTORAN process. This distinction will favor the PPCA-MV process despite the stability of the FACTORAN values.

Table 2: Sample of the reconstructed images for the 5 processes

#	Initial image	FACTORAN	ICA	PCA	PPCA	PPCA-MV
75						
20						

For PPCA and PPCA-MV, the value of the PSNR on the image number 75 is outlier but is below the mean. However, for FACTORAN, the PSNR of the reconstruction of the image 20 is outlier and is above the mean. This could be explained by the presence of contrast of black in the image 75 and the presence of fog in the picture 20 (table 2). We notice that the PCA and ICA reconstruction for the images 75 and 20 are visually not satisfying.

The samples are not Gaussian (figure 8 and table 3), but are large, with an approximately symmetrical distribution and a variance of the same size. A parametric ANOVA test can be performed to compare the 5 processes

Table 3: Processes comparison of camera 1 : digital indicators

Process	Sample size	Mean	Variance
FACTORAN	103	847,204369	311,598793
ICA	103	760,819126	3266,50409
PCA	103	836,713689	2133,71738
PPCA	103	664,608738	2275,76999
PPCA-MV	103	994,650097	3597,00824

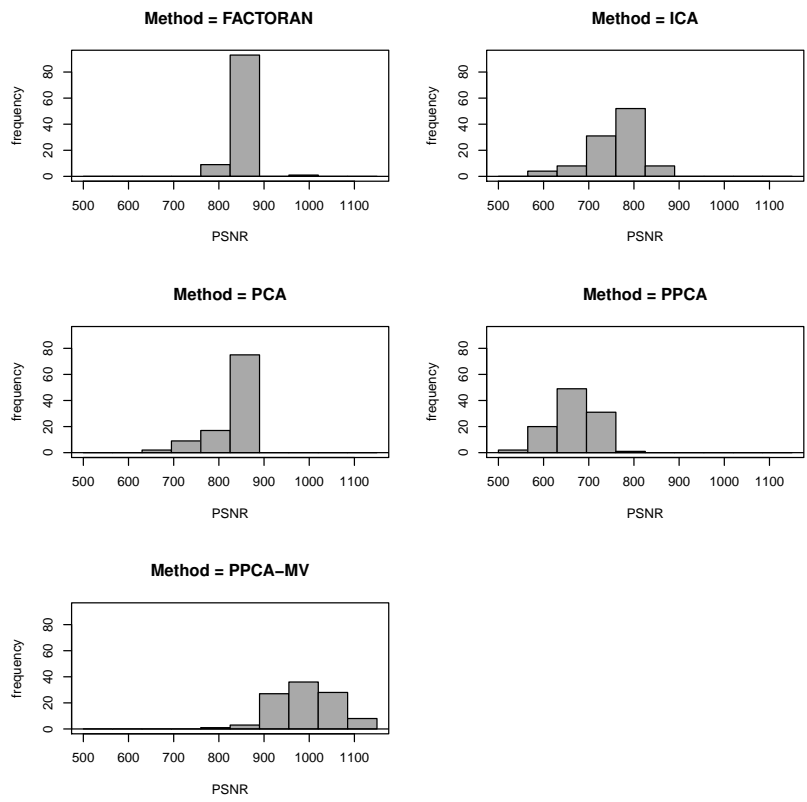


Figure 8: Processes comparison of camera 1 : histogram frequency

between them on the images from the camera 1. We test the hypothesis  $H_0$  against the hypothesis  $H_1$ :

$$\begin{cases} H_0 = \text{"equality of all the processes"} \\ H_1 = \text{"at least two processes are different"} \end{cases} \quad (16)$$

The value of the p-value is about  $4,048 \times 10^{-243}$ , so we reject  $H_0$ . **The processes are significantly different on the camera 1.** The *Student* test will allow us to test the processes 2 by 2 to derive one that allows the best image reconstruction.

Table 4: *Student* p-value and comparison of PPCA-MV process with the others for the camera 1.

Process	p-value	Comparison
PPCA-MV vs FACTORAN	$2.02258 \times 10^{-47}$	PPCA-MV is better than FACTORAN
PPCA-MV vs PCA	$4.70365 \times 10^{-52}$	PPCA-MV is better than PCA
PPCA-MV vs ICA	$2.43671 \times 10^{-73}$	PPCA-MV is better than ICA
PPCA-MV vs PPCA	$1.9757 \times 10^{-102}$	PPCA-MV is better than PPCA
<b>PPCA seems to be the best process for the reconstruction of the images from the camera 1</b>		

#### 4.2. Results : study of the PSNR measurements on the 6 cameras

Table 5: Processes's information : digital indicators

Process	Sample size	Mean	Variance	p-value test ANOVA
FACTORAN	967	889,2930196	1246,27231	$< 2 \times 10^{-16}$
PCA	967	885,2467528	2934,08331	
PPCA	967	636,1562668	3165,78782	
PPCA-MV	967	1037,886163	4815,87215	
ICA	753	792,4279017	11827,7726	

The samples are not Gaussian, but are large, roughly symmetrical distribution (histograms of figure 9) and variance of the same size (table 5). We can perform a parametric test ANOVA to compare the 5 processes together on all the images from the different cameras.  $H_0$  is tested against hypothesis  $H_1$ :

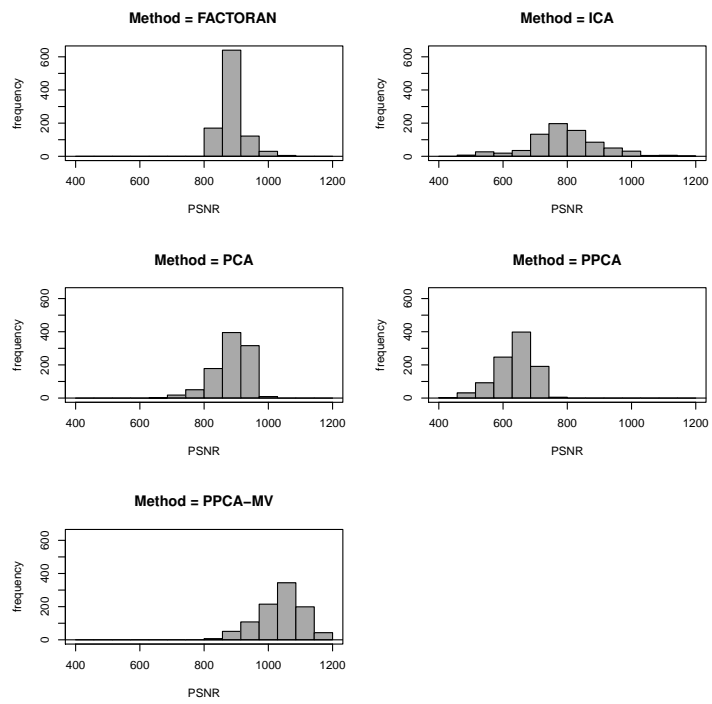


Figure 9: Frequencies histogram of image processing for the 6 cameras

$$\begin{cases} H_0 = \text{"equality of all the processes"} \\ H_1 = \text{"at least two processes are different"} \end{cases} \quad (17)$$

The value of the p-value is very low, so we reject  $H_0$ . The processes are significantly different on all the cameras. The *Student* test will allow us to test the processes 2 by 2 to derive one that allows the best image reconstruction.

Table 6: *Student* p-value and comparison of PPCA-MV process with the others for the 6 cameras

Process	p-value	Comparison
PPCA-MV vs FACTORAN	$< 2 \times 10^{-16}$	PPCA-MV is better than FACTORAN
PPCA-MV VS PCA	$< 2 \times 10^{-16}$	PPCA-MV is better than PCA
PPCA-MV vs ICA	$< 2 \times 10^{-16}$	PPCA-MV is better than ICA
PPCA-MV vs PPCA	$< 2 \times 10^{-16}$	PPCA-MV is better than PPCA
<b>PPCA-MV seems to be the best process for the reconstruction of the images from all the cameras</b>		

#### 4.3. Results : comparative summary of the processes on each camera

For each camera, the distribution is more or less symmetrical. The samples being large, the parametric test ANOVA allows us to reject the hypothesis for each camera that the processes are equals. **It nevertheless notes that the PPCA-MV and FACTORAN processes on each camera have an average very close.**

*Student's* test will allow us to test the equality of these two processes PPCA-MV and FACTORAN on images of each camera. Testing summaries are presented in the table 7.

## 5. Conclusions and future directions

One process appear to distinguish among the 5 processes to rebuild the images : PPCA-MV. Although, we can also consider FACTORAN for its variance stability.

In a further work, it would be very interesting to try the inverse problem methods and analyse sparse components. It might also help in analysing the images.

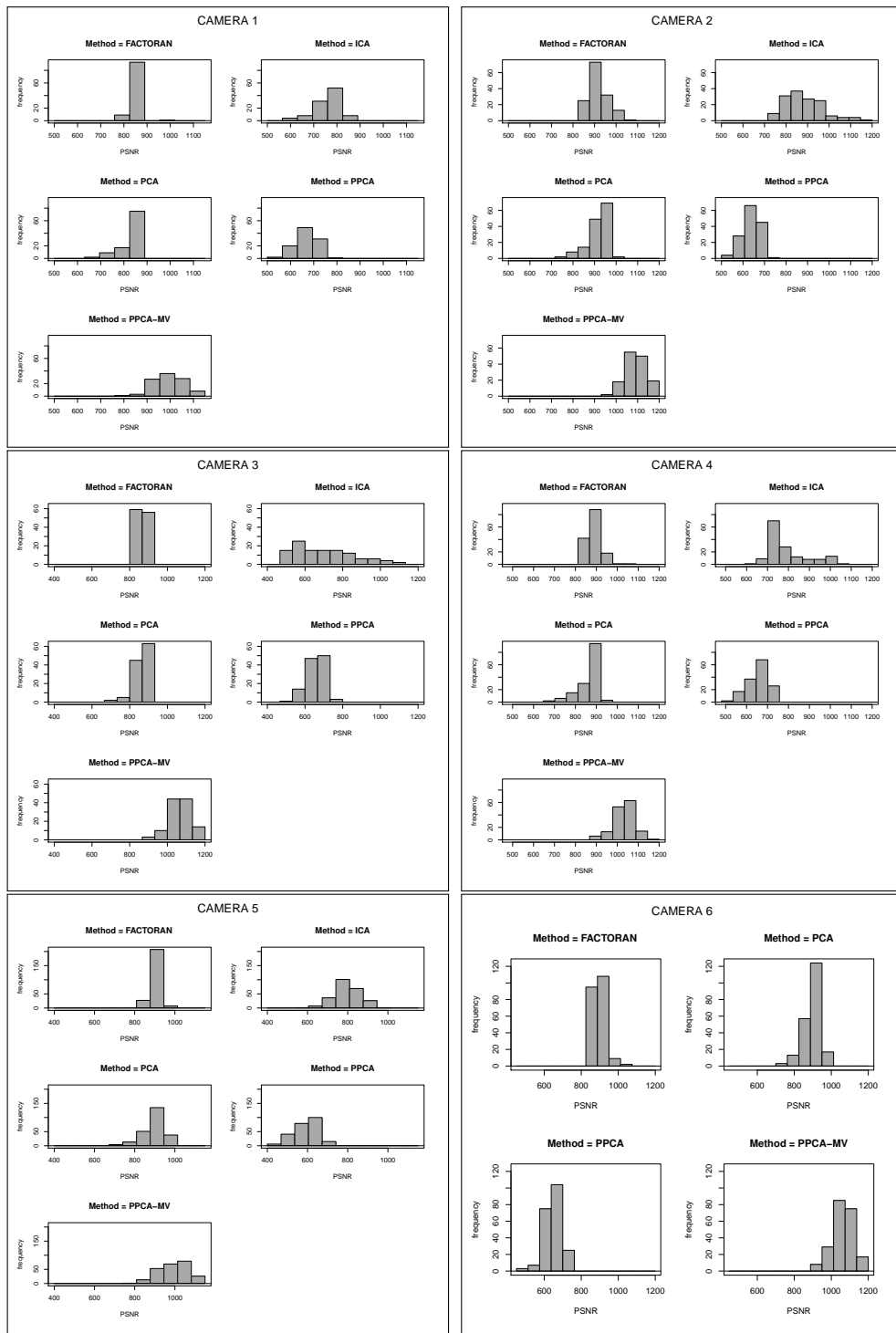


Figure 10: Frequencies histogram by camera



Table 7: Comparison of PPCA-MV and FACTORAN by camera

Camera	p-value	Comparison
Camera 1	$2,02258 \times 10^{-47}$	PPCA-MV is better than FACTORAN
Camera 2	$5,59379 \times 10^{-92}$	PPCA-MV is better than FACTORAN
Camera 3	$1,28212 \times 10^{-74}$	PPCA-MV is better than FACTORAN
Camera 4	$1,258 \times 10^{-154}$	PPCA-MV is better than FACTORAN
Camera 5	$2,41451 \times 10^{-55}$	PPCA-MV is better than FACTORAN
Camera 6	$7,4255 \times 10^{-126}$	PPCA-MV is better than FACTORAN

The *in-situ* images used for the size reduction have also their own noise : geometric variations, luminosity variations and bad weather perturbations (fog, water drop on the lense, ice, ...). Eventhough the reconstruction of the images after the principal components decomposition was not without error, we have noticed that sometimes this imperfection was because the reconstruction *deleted* the water drops on the lense, *minimized* the fog, ... In the whole workflow of image processing for analysing the evolution of the snow cover, these imperfections can be really usefull.

During our tests with Matlab, we have very often reach its limits in a space memory matter (**Error : Out of memory**). We have only taken a very small sample of out data set to try the different methods with it. Applying these technics to the whole data set will require to use a distributed and/or parallel architecture like the cloud computing.

## References

- Abdi, H., Williams, L. J., 2010. Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics 2 (4), 433–459.
- Aharon, M., Elad, M., Bruckstein, A., 2006. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. Signal Processing, IEEE Transactions on 54 (11), 4311–4322.
- Cardoso, J.-F., 2002. Analyse en composantes indépendantes. Proc. of XXXIV Journées de Statistique, Bruxelles.
- Davis, G., Mallat, S., Avellaneda, M., 1997. Adaptive greedy approximations. Constructive approximation 13 (1), 57–98.

- Faivre, R., Iooss, B., Mahévas, S., Makowski, D., Monod, H., 2013. Analyse de sensibilité et exploration de modèles: application aux sciences de la nature et de l'environnement. Editions Quae.
- Hore, A., Ziou, D., Aug 2010. Image quality metrics: PSNR vs. SSIM. In: Pattern Recognition (ICPR), 2010 20th International Conference on. pp. 2366–2369.
- Jöreskog, K. G., 1967. Some contributions to maximum likelihood factor analysis. *Psychometrika* 32 (4), 443–482.
- Liu, J.-K., Wu, H.-C., Shih, T.-Y., 2005. Effects of jpeg2000 on the information and geometry content of aerial photo compression. *Photogrammetric Engineering & Remote Sensing* 71 (2), 157–167.
- Park, T., Casella, G., 2008. The bayesian lasso. *Journal of the American Statistical Association* 103 (482), 681–686.
- Porwik, P., Lisowska, A., 2004. The haar-wavelet transform in digital image processing: its status and achievements. *Machine graphics and vision* 13 (1/2), 79–98.
- Pratt, W. K., 2002. Digital image processing : piks inside, 4th Edition. Wiley.
- Saporta, G., 1990. Probabilités, analyse des données et statistique. Editions Technip.
- Shlens, J., 2014. A tutorial on principal component analysis. CoRR abs/1404.1100.  
URL <http://arxiv.org/abs/1404.1100>
- Talukder, K. H., Harada, K., 2010. Haar wavelet based approach for image compression and quality assessment of compressed image. *International Journal of Applied Mathematics* 36, 1–9.
- Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* 58 (1), pp. 267–288.  
URL <http://www.jstor.org/stable/2346178>

- Tipping, M. E., Bishop, C. M., 1999. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61 (3), 611–622.
- Verbeek, J., 2009. Notes on probabilistic pca with missing values. Tech. rep., Tech. report.
- Wang, J. Z., 2001. Wavelets and imaging informatics: A review of the literature. *Journal of Biomedical Informatics* 34 (2), 129–141.

## 11.4 Conclusion partielle

Dans la même lancée que les différents travaux de Jean Corbel, ce travail s'inscrit dans un contexte de pluridisciplinarité. En effet, les différentes méthodes mathématiques et probabilistes appliquées à la réduction de dimensions peuvent être un véritable atout face à la masse grandissante de données récoltées, autant en nombre d'images qu'en taille.

De plus, suite à nos premiers tests effectués sur un jeu de données faible (environ 100 photos), nous avons pu remarquer un autre résultat très intéressant pour le *workflow* de traitement des photos. En effet, comme nous avons pu le voir dans le chapitre 5, le traitement de nettoyage d'une image reste un verrou scientifique. Dans notre cas particulier d'images capturées régulièrement et fréquemment, la reconstruction des images suites à une réduction de données par méthode d'analyse factorielle pourrait être une solution envisageable.



# Conclusion générale

Most people say that it is the intellect  
which makes a great scientist. They are  
wrong : it is character.

---

*Albert Einstein*

## Conclusion

L'analyse des dynamiques environnementales, et notamment l'étude de l'impact du changement climatique contemporain sur les glaciers de l'Arctique, nécessite une acquisition régulière et fréquente de données. Grâce à l'héritage des différentes missions effectuées au Spitsberg, initiées par Jean Corbel puis reprises par Thierry Brossard et Daniel Joly et les différentes équipes scientifiques, la presqu'île de Brøgger et plus particulièrement le glacier du Lovén Est a été étudié. À travers le programme HSF, le glacier a été analysé sous différents aspects : le changement climatique récent, la dynamique du manteau neigeux et suivi glaciologique, le suivi hydrologique du bassin versant du Lovén Est et la géomatique pour quantifier la réaction du glacier face aux changements climatiques. Ce programme a permis d'acquérir et d'extraire une grande quantité d'informations afin de comprendre le fonctionnement du glacier et d'approximer son évolution.

Parmi les différentes analyses menées, celle de l'évolution de la couverture neigeuse a nécessité et engendré une grande quantité de données qui n'ont pas été exploitées à leur maximum. En effet, l'analyse de la dynamique du manteau neigeux requiert une acquisition régulière et fréquente de l'état du glacier, afin de pouvoir déterminer l'évolution de la ligne d'équilibre entre la zone d'accumulation et la zone d'ablation. Malgré la qualité des images satellites existantes, leur coût et les perturbations météorologiques limite le nombre d'acquisition à environ 11 par saison. Afin de compléter l'imagerie satellitaire, un ensemble d'appareils photo automatiques et autonomes a été mis en place tout autour du bassin versant, assurant une couverture visuelle de plus de 95% du glacier et capturant 3 photos par jour.

Même si ces appareils photo ont permis de générer une grande quantité de photos *in situ* à exploiter, les contraintes météorologiques se sont transformées en bruit sur les photos perturbées (goutte d'eau, brouillard, etc.). Le traitement nécessaire des photos *in situ* afin de résoudre la problématique des perturbations et d'obtenir un équivalent de la vue satellite permettant une étude de l'évolution du manteau neigeux, est complexe et varié. La complexité est telle que l'analyse doit être manuelle, et donc seulement une petite portion de la masse de données générée est traitée. Seule l'informatique est à même de traiter l'intégralité des données.

Depuis la naissance de l'informatique, puis avec le développement rapide et important d'Internet, tout comme l'évolution des composants électroniques (processeurs, mémoire, etc.), plusieurs concepts et architectures de stockage et traitement de l'information ont émergé. Le *cloud computing* fait partie de ces dernières structures émergentes. Il offre à ses utilisateurs une capacité de traitement et de stockage ressemblant à l'infini et un ensemble de fonctionnalités ou services qui facilite sa gestion et son utilisation. En plus de son architecture distribuée, le *cloud computing* force une nouvelle manière de penser dans le développement d'outil *cloud*, autant dans le stockage de données que dans le traitement de celles-ci.

Bien que l'émergence du *cloud computing* ait été acceptée dans notre utilisation courante (Google Maps, Google Drive, Amazon, etc), cela l'est beaucoup moins dans le domaine des sciences environnementales. Au travers de cette thèse, nous nous sommes posés trois questions principales :

1. Comment automatiser les traitements tout en restant assez générique afin de pouvoir traiter n'importe quelle image *in situ* (provenance et perturbations) ?
2. En quoi le *cloud computing* permet d'automatiser les traitements ? En quoi est-il plus intéressant qu'une simple machine ou qu'un serveur de calcul ?
3. Face à la quantité de données, peut-on réduire les données à stocker sans pour autant perdre de l'information ?

À travers le modèle d'architecture présenté dans l'article [Ranisavljević et al., 2014a], et avec la partie d'orchestration de services plus détaillée dans l'article [Ranisavljević et al., 2013], nous avons montré la possibilité d'automatisation des traitements d'images. En effet, la scalabilité du *cloud* nous a permis de prendre en charge l'ensemble des photos *in situ*, et la conférence [Ranisavljević et al., 2014b] montre l'intérêt d'utiliser une telle architecture qui se déploie automatiquement afin d'absorber la charge supplémentaire. La modularité du *cloud* nous a permis de rendre le modèle architectural dynamique et adaptatif aux changements (ajout, suppression, mise à jour) de services pour le traitement d'images. Couplé à l'ordonnanceur de services, l'évolution des services peut être pris en compte directement pendant l'exécution de la chaîne de traitement. Le découpage du traitement d'images *in situ* en plusieurs tâches "unitaires" et implémentées en tant que web services nous a permis de rester générique aux données traitées. En effet, une bonne partie des traitements (projection, mosaïque, classification, etc) ne devrait pas varier si le paysage étudié change.

Quant à la réduction de dimensions des données, les premiers tests effectués par les méthodes d'analyse factorielle et de décomposition en composantes principales ou indépendantes présentent des résultats prometteurs. Nous avons montré qu'il est possible de réduire les données à stocker, puis de les reconstruire, et de retrouver l'information sans trop d'erreurs.

## Perspectives

Ce travail de thèse peut être considéré comme étant dans le commencement d'un potentiel interdisciplinaire entre l'informatique et la géographie. Malgré les premières pierres posées, il reste un certain nombre d'aspects qui seraient intéressants à poursuivre.

**Le traitement des données :** La chaîne de traitements des photos *in situ* pour l'analyse de l'évolution de la couverture neigeuse n'est pas pour le moment opérationnelle puisque certains services sont manquants ou semi-automatisés.

- Après avoir chargé les photos *in situ* en base de données, il est important de pouvoir déterminer automatiquement si une image est utilisable. Dans le cas des images partiellement perburgées (par exemple, par une goutte d'eau) qui doivent être "nettoyées", il est semble que la recomposition après réduction des données donne un résultat concluant.
- Même si les appareils photo ont évolué au cours du temps pour ralentir le temps de décharge de la batterie et limiter les défaillances, la perte de la date des photos liée au déchargement de la pile interne reste un problème important, réduisant le nombre d'images utilisables pour l'analyse de l'évolution du manteau neigeux.
- Les traitements semi-automatiques comme le recadrage des photos nécessitent, pour le moment, un fichier de configuration définissant, pour chaque période de temps, les points de contrôle associés à la transformation. Automatiser entièrement le traitement correspondrait à utiliser des méthodes statistiques et de *machine learning* afin de déterminer s'il y a eu un changement de cadrage de l'appareil photo puis de déterminer des points de contrôle (ou autres marqueurs similaires) afin de pouvoir appliquer la transformation. Cette étude reste complexe à mettre en place, tout particulièrement à cause de la paramétrisation de ces différentes méthodes face aux variations météorologiques qui viennent parasiter l'image à traiter.

**Le Big Data :** Même si cette thèse n'a pas été abordée comme étant un problème de *Big Data* puisque le volume de données géré n'était pas suffisant, l'évolution et l'ouverture du marché de la télédétection va nous y mener. De nouvelles problématiques concernant le stockage et le traitement de ces données si importantes émergent. C'est dans ce cadre de recherche reliant le *cloud computing*, le *big data* et les géosciences que le projet TORUS (*Toward an Open Resources Upon Services : Cloud of environmental data*)<sup>1</sup> a été mis en place depuis 2015. Ce projet *Erasmus + Capacity Building*, regroupant des scientifiques de France, Belgique, Italie, Viêt Nam et Thaïlande, a pour objectif de développer la recherche sur le cloud computing dans les sciences de l'environnement et promouvoir son éducation dans les pays des partenaires d'Asie du Sud-Est.

**La réduction des données :** Les premières analyses que nous avons pu mener sur l'utilisation des composantes principales et indépendantes nous montrent qu'il y a une possibilité de réduction de dimensions. Nous nous sommes focalisés sur une analyse factorielle, mais il serait intéressant de regarder les autres méthodes probabilistes et statistiques. De plus, outre l'avantage de la réduction de dimensions, l'analyse factorielle nous a fait découvrir une nouvelle possibilité pour le traitements de photos *in situ* lors de la reconstruction du mélange. Pendant notre étude, nous nous sommes concentrés sur la possibilité de régénérer les données initiales (le mélange) à partir des composantes (la source). Il serait également intéressant d'analyser les composantes. Elles pourraient, comme la reconstruction, nous dévoiler d'autres aspects nécessaires pour le traitement des images *in situ*.

---

1. Détail du projet TORUS sur : <http://www.cloud-torus.com>





# Annexe



# Bibliographie

- [Abiteboul, 1997] Abiteboul, S. (1997). Querying semi-structured data. In *International Conference on Database Theory*, pages 1–18. Springer.
- [Agassiz and Bettannier, 1840] Agassiz, L. and Bettannier, J. (1840). *Etudes sur les glaciers*. Jent et Gassmann.
- [Ahuja et al., 1993] Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). Network flows : theory, algorithms, and applications.
- [Alonso et al., 2003] Alonso, G., Casati, F., Kuno, H., and Machiraju, V. (2003). *Web Services - Concepts, Architectures and Applications*. Springer, 1 edition.
- [Anderson et al., 1995] Anderson, T. E., Culler, D. E., and Patterson, D. A. (1995). The berkeley networks of workstations (now) project. In *Compton'95. Technologies for the Information Superhighway', Digest of Papers.*, pages 322–326. IEEE.
- [André, 1986] André, M.-F. (1986). Dating slope deposits and estimating rates of rock wall retreat in northwest spitsbergen by lichenometry. *Geografiska Annaler. Series A. Physical Geography*, pages 65–75.
- [Antoniou and Van Harmelen, 2004] Antoniou, G. and Van Harmelen, F. (2004). Web ontology language : Owl. In *Handbook on ontologies*, pages 67–92. Springer.
- [Armbrust et al., 2010] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., and Zaharia, M. (2010). A view of cloud computing. *Commun. ACM*, 53(4) :50–58.
- [Arora and Aggarwal, 2013] Arora, R. and Aggarwal, R. R. (2013). Modeling and querying data in mongodb. *International Journal of Scientific and Engineering Research*, 4 :141–144.
- [Austin et al., 2004] Austin, D., Barbir, A., Peters, E., and Ross-Talbot, S. (2004). Web services choreography requirements. W3C Working Draft. <https://www.w3.org/TR/ws-chor-reqs/>.
- [Baader, 2003] Baader, F. (2003). *The description logic handbook : Theory, implementation and applications*. Cambridge university press.
- [Bailly et al., 2016] Bailly, A., Beguin, H., and Scariati, R. (2016). *Introduction à la géographie humaine-9e éd.* Armand Colin.
- [Barnett et al., 2005] Barnett, T. P., Adam, J. C., and Lettenmaier, D. P. (2005). Potential impacts of a warming climate on water availability in snow-dominated regions. *Nature*, 438(7066) :303–309.
- [Barry et al., 2011] Barry, R., Jania, J., and Birkenmajer, K. (2011). Ab dobrowolski—the first cryospheric scientist—and the subsequent development of cryospheric science. *Hist. Geo-Space Sci*, 2(1) :75–79.

- [Beauche and Poizat, 2008] Beauche, S. and Poizat, P. (2008). Automated service composition with adaptive planning. In Bouguettaya, A., Krueger, I., and Margaria, T., editors, *Service-Oriented Computing – ICSOC 2008*, volume 5364 of *Lecture Notes in Computer Science*, pages 530–537. Springer Berlin Heidelberg.
- [Becker et al., 1995] Becker, D. J., Sterling, T., Savarese, D., Dorband, J. E., Ranawak, U. A., and Packer, C. V. (1995). Beowulf : A parallel workstation for scientific computation. In *Proceedings, International Conference on Parallel Processing*, volume 95.
- [Beckman, 2005] Beckman, P. H. (2005). Building the teragrid. *Philosophical Transactions of the Royal Society of London A : Mathematical, Physical and Engineering Sciences*, 363(1833) :1715–1728.
- [Bellwood et al., 2004] Bellwood, T., Steve, C., Clément, L., Colgrave, J., Dovey, M. J., Feygin, D., Hatley, A., Kochman, R., Macias, P., Novotny, M., Paolucci, M., von Riegen, C., Rogers, T., Sycara, K., Wenzel, P., and Wu, Z. (2004). Uddi version 3.0. UDDI Spec Technical Committee Draft. [http://www.uddi.org/pubs/uddi\\_v3.htm](http://www.uddi.org/pubs/uddi_v3.htm).
- [Benatallah et al., 2005] Benatallah, B., Dijkman, R. M., Dumas, M., and Maamar, Z. (2005). *Service-Oriented Software System Engineering : Challenges and Practices*, chapter Service Composition : Concepts, Techniques, Tools and Trends, pages 48–66. IGI Global.
- [Bernard, 2011] Bernard, E. (2011). *Les dynamiques spatio-temporelles d’un petit hydrosystème arctique : approche nivo-glaciologique dans un contexte de changement climatique contemporain (bassin du glacier Austre Lovén, Spitsberg, 79° N)*. PhD thesis, Université de Franche-Comté.
- [Bernard et al., 2014] Bernard, E., Friedt, J.-M., Saintenoy, A., Tolle, F., Griselin, M., and Marlin, C. (2014). Where does a glacier end ? gpr measurements to identify the limits between valley slopes and actual glacier body. application to the austre lovénbreen, spitsbergen. *International Journal of Applied Earth Observation and Geoinformation*, 27 :100–108.
- [Bernard et al., 2013a] Bernard, E., Tolle, F., Friedt, J.-M., Marlin, C., and Griselin, M. (2013a). Five consecutive years of mass balance observations to understand glacier reaction to present climate change (austre lovénbreen, svalbard 79° n). In *International Arctic Science Committee workshop on the dynamics and mass budget of Arctic glaciers*.
- [Bernard et al., 2009] Bernard, E., Tolle, F., Griselin, M., Laffly, D., and Marlin, C. (2009). Quantification des hauteurs de neige et des températures de l’air à la surface d’un glacier : du terrain à l’interpolation, confrontation de méthodes. In *Actes des Neuvièmes Rencontres de Théo Quant*, Besançon, France.
- [Bernard et al., 2013b] Bernard, É., Friedt, J., Tolle, F., Griselin, M., Martin, G., Laffly, D., and Marlin, C. (2013b). Monitoring seasonal snow dynamics using ground based high resolution photography (austre lovénbreen, svalbard, 79°n). *ISPRS Journal of Photogrammetry and Remote Sensing*, 75 :92–100.
- [Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., Lassila, O., et al. (2001). The semantic web. *Scientific american*, 284(5) :28–37.
- [Bertrand, 1972] Bertrand, G. (1972). La «science du paysage», une «science diagonale». *Revue géographique des Pyrénées et du Sud-Ouest*, 43(2) :127–134.
- [Besnier and Ladurie, 1968] Besnier, R. and Ladurie, E. L. (1968). Histoire du climat depuis l’an mil.

- [Bezdek et al., 1981] Bezdek, J. C., Coray, C., Gunderson, R., and Watson, J. (1981). Detection and characterization of cluster substructure i. linear structure : Fuzzy c-lines. *SIAM Journal on Applied Mathematics*, 40(2) :339–357.
- [Bhardwaj et al., 2010] Bhardwaj, S., Jain, L., and Jain, S. (2010). Cloud computing : A study of infrastructure as a service (iaas). *International Journal of engineering and information Technology*, 2(1) :60–63.
- [Bieber and Carpenter, 2001] Bieber, G. and Carpenter, J. (2001). Introduction to service-oriented programming (rev 2.1). In *OpenWings Whitepaper*.
- [Bishop, 1995] Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford university press.
- [Boas, 1887] Boas, F. (1887). The study of geography. *Science*, pages 137–141.
- [Bourdon, 2007] Bourdon, J. (2007). Multi-agent systems for the automatic composition of semantic web services in dynamic environments. Master’s thesis, University of Saint-Etienne.
- [Brewer, 2012] Brewer, E. (2012). Cap twelve years later : How the " rules " have changed. *Computer*, 45(2) :23–29.
- [Brickley and Guha, 2014] Brickley, D. and Guha, R. (2014). Rdf schema 1.1. W3C Recommendation. [http ://www.w3.org/TR/rdf-schema](http://www.w3.org/TR/rdf-schema).
- [Brossard and Joly, 1994] Brossard, T. and Joly, D. (1994). Probability models, remote sensing and field observation : test for mapping some plant distributions in the kongsfjord area, svalbard. *Polar research*, 13(1) :153–161.
- [Buneman, 1997] Buneman, P. (1997). Semistructured data. In *Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 117–121. ACM.
- [Buneman et al., 2001] Buneman, P., Fan, W., and Weinstein, S. (2001). Constraints for semistructured data. *optimization*, 16(13) :21.
- [Castelfranchi, 1994] Castelfranchi, C. (1994). Guarantees for autonomy in cognitive agent architecture. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 56–70. Springer.
- [Cattell, 2011] Cattell, R. (2011). Scalable sql and nosql data stores. *Acm Sigmod Record*, 39(4) :12–27.
- [Chang et al., 2008] Chang, F., Dean, J., Ghemawat, S., Hsieh, W. C., Wallach, D. A., Burrows, M., Chandra, T., Fikes, A., and Gruber, R. E. (2008). Bigtable : A distributed storage system for structured data. *ACM Transactions on Computer Systems (TOCS)*, 26(2) :4.
- [Chapin et al., 2005] Chapin, F. S., Sturm, M., Serreze, M., McFadden, J., Key, J., Lloyd, A., McGuire, A., Rupp, T., Lynch, A., Schimel, J., et al. (2005). Role of land-surface changes in arctic summer warming. *science*, 310(5748) :657–660.
- [Chapman and Walsh, 1993] Chapman, W. L. and Walsh, J. E. (1993). Recent variations of sea ice and air temperature in high latitudes. *Bulletin of the American Meteorological Society*, 74(1) :33–47.
- [Charif, 2007] Charif, Y. (2007). *Chorégraphie dynamique de services basée sur la coordination d’agents introspectifs*. PhD thesis, Laboratoire d’Informatique de Paris 6.
- [Chien et al., 1997] Chien, A. A., Pakin, S., Lauria, M., Buchanan, M., Hane, K., Giannini, L. A., and Prusakova, J. (1997). High performance virtual machines (hpvm’s) : Clusters with super-computing api’s and performance. In *PPSC*.

- [Chodorow, 2011] Chodorow, K. (2011). *Scaling MongoDB*. " O'Reilly Media, Inc."
- [Christensen et al., 2001] Christensen, E., Curbera, F., Meredith, G., Weerawarana, S., et al. (2001). Web services description language (wsdl) 1.1. W3C Note. <http://www.w3.org/TR/wsdl>.
- [Codd, 1970] Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13(6) :377–387.
- [Codd, 1979] Codd, E. F. (1979). Extending the database relational model to capture more meaning. *ACM Transactions on Database Systems (TODS)*, 4(4) :397–434.
- [Codd, 1982] Codd, E. F. (1982). Relational database : a practical foundation for productivity. *Communications of the ACM*, 25(2) :109–117.
- [Corbel, 1953] Corbel, J. (1953). Problèmes de morphologie glaciaire au spitzberg. *Revue de géographie de Lyon*, 28(3) :262–268.
- [Corbel, 1961] Corbel, J. (1961). Morphologie périglaciaire dans l'arctique. *Annales de Géographie*, 70(377) :1–24.
- [Corbel, 1966] Corbel, J. (1966). La base du cnrs au spitzberg. les premiers résultats. *Bulletin de l'Association de géographes français*, 43(342-343) :2–10.
- [Corpetti, 2011] Corpetti, T. (2011). *Images & télédétection : analyse de séquences à basse et très haute résolution spatiale*. Hdr, Université Rennes 1.
- [Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20 :273–297.
- [Curbera et al., 2001] Curbera, F., Nagy, W., and Weerawarana, S. (2001). Web services : Why and how. In *OOPSLA 2001 Workshop on Object-Oriented Web Services*, pages 249–259.
- [de La Blache, 1896] de La Blache, P. V. (1896). Le principe de la géographie générale. *Annales de Géographie*, 5(20) :129–142.
- [DeCandia et al., 2007] DeCandia, G., Hastorun, D., Jampani, M., Kakulapati, G., Lakshman, A., Pilchin, A., Sivasubramanian, S., Vosshall, P., and Vogels, W. (2007). Dynamo : amazon's highly available key-value store. *ACM SIGOPS Operating Systems Review*, 41(6) :205–220.
- [Dede et al., 2013] Dede, E., Govindaraju, M., Gunter, D., Canon, R. S., and Ramakrishnan, L. (2013). Performance evaluation of a mongodb and hadoop platform for scientific data analysis. In *Proceedings of the 4th ACM workshop on Scientific cloud computing*, pages 13–20. ACM.
- [Delangle et al., 2010] Delangle, E., Marlin, C., Griselin, M., Laffly, D., Bernard, E., and Friedt, J. M. (2010). Hydrochemical response of a polar glacier facing the recent climate changes (austrelovenbre, svalbard, 79° n). In *3rd European Conference on Permafrost*.
- [Dematteis, 1971] Dematteis, G. (1971). Les relations entre types d'espaces différents en tant qu'objet de la géographie théorique. In *Cahiers de Géographie de Besançon*, volume 2.
- [Dillon et al., 2010] Dillon, T., Wu, C., and Chang, E. (2010). Cloud computing : issues and challenges. In *2010 24th IEEE international conference on advanced information networking and applications*, pages 27–33. Ieee.
- [Dornemann et al., 2009] Dornemann, T., Juhnke, E., and Freisleben, B. (2009). On-demand resource provisioning for bpel workflows using amazon's elastic compute cloud. In *Cluster Computing and the Grid, 2009. CCGRID '09. 9th IEEE/ACM International Symposium on*, pages 140–147.

- [Dowdeswell et al., 1997] Dowdeswell, J. A., Hagen, J. O., Björnsson, H., Glazovsky, A. F., Harrison, W. D., Holmlund, P., Jania, J., Koerner, R. M., Lefauconnier, B., Ommanney, C. S. L., et al. (1997). The mass balance of circum-arctic glaciers and recent climate change. *Quaternary research*, 48(1) :1–14.
- [Dreiseitl and Ohno-Machado, 2002] Dreiseitl, S. and Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models : a methodology review. *Journal of biomedical informatics*, 35(5) :352–359.
- [Duda et al., 2000] Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition.
- [Erl, 2008] Erl, T. (2008). *Soa : principles of service design*, volume 1. Prentice Hall Upper Saddle River.
- [Erol et al., 1994] Erol, K., Hendler, J., and Nau, D. S. (1994). Htn planning : Complexity and expressivity. In *AAAI*, volume 94, pages 1123–1128.
- [Farrell and Lausen, 2007] Farrell, J. and Lausen, H. (2007). Semantic annotations for wsd and xml schema. W3C Recommendation. <http://www.w3.org/TR/sawsdl/>.
- [Fenstermacher et al., 2006] Fenstermacher, D., Street, C., McSherry, T., Nayak, V., Overby, C., and Feldman, M. (2006). The cancer biomedical informatics grid (cabig tm). In *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pages 743–746. IEEE.
- [Ferber and Perrot, 1995] Ferber, J. and Perrot, J.-F. (1995). *Les systèmes multi-agents : vers une intelligence collective*. InterEditions.
- [Foster et al., 2001] Foster, I., Kesselman, C., and Tuecke, S. (2001). The anatomy of the grid : Enabling scalable virtual organizations. *International journal of high performance computing applications*, 15(3) :200–222.
- [Foster et al., 2008] Foster, I., Zhao, Y., Raicu, I., and Lu, S. (2008). Cloud computing and grid computing 360-degree compared. In *Grid Computing Environments Workshop, 2008. GCE '08*, pages 1–10.
- [Francou and Vincent, 2007] Francou, B. and Vincent, C. (2007). *Les glaciers à l'épreuve du climat*. IRD Editions.
- [Freund et al., 1999] Freund, Y., Schapire, R., and Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780) :1612.
- [Friedt et al., 2011] Friedt, J.-M., Tolle, F., Bernard, E., Griselin, M., Laffly, D., and Marlin, C. (2011). Assessing the relevance of digital elevation models to evaluate glacier mass balance : application to austre lovénbreen (spitsbergen, 79°n). *Polar Record*, 48 :2–10.
- [Garcia-Molina et al., 2008] Garcia-Molina, H., Ullman, J. D., and Widom, J. (2008). *Database systems : the complete book*. Pearson Education Inc.
- [Gardner and Sharp, 2007] Gardner, A. S. and Sharp, M. (2007). Influence of the arctic circumpolar vortex on the mass balance of canadian high arctic glaciers. *Journal of Climate*, 20(18) :4586–4598.
- [Gath and Geva, 1989] Gath, I. and Geva, A. B. (1989). Unsupervised optimal fuzzy clustering. *IEEE Transactions on pattern analysis and machine intelligence*, 11(7) :773–780.
- [Geelan et al., 2008] Geelan, J. et al. (2008). Twenty-one experts define cloud computing. *Cloud Computing Journal*, 4 :1–5.



- [Genesereth and Ketchpel, 1994] Genesereth, M. R. and Ketchpel, S. P. (1994). Software agents. *Commun. ACM*, 37(7) :48–53.
- [Geoffray, 1968] Geoffray, H. (1968). *Étude du bilan hydrologique et de l'érosion sur un bassin partiellement englacé, Spitsberg, Baie du Roi, 79 Lat. Nord*. PhD thesis, Thèse de 3e cycle, Université de Rennes.
- [Gerland et al., 2008] Gerland, S., Renner, A., Godtliessen, F., Divine, D., and Løyning, T. (2008). Decrease of sea ice thickness at hopen, barents sea, during 1966–2007. *Geophysical Research Letters*, 35(6).
- [Gilbert and Lynch, 2012] Gilbert, S. and Lynch, N. A. (2012). Perspectives on the cap theorem. In *Institute of Electrical and Electronics Engineers*. Institute of Electrical and Electronics Engineers.
- [Gillman, 1985] Gillman, D. (1985). Triangulations for rubber sheeting. In *Proceedings of 7th International Symposium on Computer Assisted Cartography (AutoCarto 7)*, volume 199.
- [Glover and Kochenberger, 2006] Glover, F. W. and Kochenberger, G. A. (2006). *Handbook of metaheuristics*, volume 57. Springer Science & Business Media.
- [Griselin, 1982] Griselin, M. (1982). *Les modalités de l'écoulement liquide et solide sur les marges polaires (exemple du bassin Loven Est, côte NW du Spitsberg)*. PhD thesis, Thèse, Université de Nancy II.
- [Griselin, 2010] Griselin, M. (2010). Projet SENSOR-FLOWS. Technical report, Centre National de la Recherche Scientifique.
- [Griselin and Marlin, 1999] Griselin, M. and Marlin, C. (1999). Approche quantitative et géochimique du gradient altitudinal des précipitations sur un bassin versant partiellement englacé sur la côte occidentale du spitsberg. *La Houille Blanche*, 5(5) :34–39.
- [Griselin et al., 1995] Griselin, M., Marlin, C., Dever, L., and Moreau, L. (1995). Hydrologie et géochimie du glacier loven est, spitsberg. In *3e symposium international, cavités glaciaires et cryokarst en régions polaires et de haute montagne : Chamonix-France, 1er-6 novembre 1994 : actes*, volume 561, page 61. Presses Univ. Franche-Comté.
- [Griselin et al., 2009a] Griselin, M., Marlin, C., Laffly, D., Bernard, E., and Delangle, E. (2009a). Forty years of weather data to understand recent climate change in the arctic (svalbard, 79° n). In *IOP Conference Series : Earth and Environmental Science*, volume 6, page 012009. IOP Publishing.
- [Griselin et al., 2009b] Griselin, M., Marlin, C., Laffly, D., Bernard, E., Delangle, E., Friedt, J.-M., Schwamborn, G., and Tolle, F. (2009b). Hydro-glaciology research on the austre lovenbreen since the 60s to the last ipy. In *Svalbard Cooperation Workshop*, volume 17, page 22.
- [Griselin et al., 2009c] Griselin, M., Marlin, C., Laffly, D., Bernard, E., Delangle, E., Friedt, J.-M., Schwamborn, G., and Tolle, F. (2009c). Recent climate change observed in svalbard since fourty years in ny alesund (79° n). In *MOCA 09, Our warming Planet, Joint assembly, symposium J01 Observations of High Latitude Climate Change*, Montréal.
- [Griselin et al., 2007] Griselin, M., Marlin, C., Laffly, D., Schwarmborn, D., Friedt, J.-M., and Martin, G. (2007). The austrelovenbre (79° n) : a glacier under hydrological survey. In *Ny Alesund and IPY*.
- [Griselin et al., 2010] Griselin, M., Martin, C., Laffly, D., Bernard, E., Delangle, E., Friedt, J.-M., Martin, G., Saintenoy, A., Tolle, F., Ormaux, S., Nageleisen, S., and Schwamborn, G. (2010). GDR mutations polaires : sociétés et environnement. Technical report, CNRS Département des Sciences Humaines et Sociales.

- [Gross and Yellen, 2004] Gross, J. L. and Yellen, J. (2004). *Handbook of graph theory*. CRC press.
- [Gudgin et al., 2007] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J.-J., Nielsen, H. F., Karmarkar, A., and Lafon, Y. (2007). Soap version 1.2. W3C Recommendation. <http://www.w3.org/TR/soap12/>.
- [Gustafson and Kessel, 1978] Gustafson, D. and Kessel, W. (1978). Fuzzy clustering with a fuzzy covariance matrix. In *IEEE CDC*, pages 761–766.
- [Güting, 1994] Güting, R. H. (1994). Graphdb : A data model and query language for graphs in databases. In *Proc. 20th Int. Conf. on Very Large Data Bases*, pages 297–308.
- [Haerder and Reuter, 1983] Haerder, T. and Reuter, A. (1983). Principles of transaction-oriented database recovery. *ACM Computing Surveys (CSUR)*, 15(4) :287–317.
- [Hassen, 2009] Hassen, R. R. (2009). *Automatic composition of protocol-based Web services*. PhD thesis, Université Blaise Pascal-Clermont-Ferrand II.
- [Hassenzahl et al., 2000] Hassenzahl, M., Platz, A., Burmester, M., and Lehner, K. (2000). Hedonic and ergonomic quality aspects determine a software’s appeal. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 201–208. ACM.
- [Hecht and Jablonski, 2011] Hecht, R. and Jablonski, S. (2011). Nosql evaluation : a use case oriented survey. In *International Conference on Cloud and Service Computing*.
- [Hilley, 2009] Hilley, D. (2009). Cloud computing : A taxonomy of platform and infrastructure-level offerings. Technical report, Georgia Institute of Technology.
- [Hinkler et al., 2003] Hinkler, J., Ørbæk, J. B., and Hansen, B. U. (2003). Detection of spatial, temporal, and spectral surface changes in the ny-ålesund area 79 n, svalbard, using a low cost multispectral camera in combination with spectroradiometer measurements. *Physics and Chemistry of the Earth, Parts A/B/C*, 28(28) :1229–1239.
- [Hinkler et al., 2002] Hinkler, J., Petersen, S. B., Rasch, M., and Hansen, B. U. (2002). Automatic snow cover monitoring at high temporal and spatial resolution, using images taken by a standard digital camera. *International Journal of Remote Sensing*, 23(21) :4669 – 4682.
- [Hinzman et al., 2005] Hinzman, L. D., Bettez, N. D., Bolton, W. R., Chapin, F. S., Dyurgerov, M. B., Fastie, C. L., Griffith, B., Hollister, R. D., Hope, A., Huntington, H. P., et al. (2005). Evidence and implications of recent climate change in northern alaska and other arctic regions. *Climatic Change*, 72(3) :251–298.
- [Höfer and Karagiannis, 2011] Höfer, C. and Karagiannis, G. (2011). Cloud computing services : taxonomy and comparison. *Journal of Internet Services and Applications*, 2(2) :81–94.
- [Holland et al., 2010] Holland, M. M., Serreze, M. C., and Stroeve, J. (2010). The sea ice mass budget of the arctic and its future change as simulated by coupled climate models. *Climate Dynamics*, 34(2-3) :185–200.
- [Huhns, 2002] Huhns, M. N. (2002). Agents as web services. *IEEE Internet computing*, 6(4) :93–95.
- [IPCC, 1995] IPCC (1995). Climate change 1995 : The science of climate change. Technical report, Cambridge University Press.
- [IPCC, 2013] IPCC (2013). Climate change 2013 : The physical science basis. Technical report, Cambridge University Press, IPCC, Geneva, Switzerland.
- [IPCC, 2014] IPCC (2014). Climate change 2014 : Synthesis report. Technical report, Cambridge University Press, IPCC, Geneva, Switzerland.

- [Jacobson et al., 1999] Jacobson, I., Booch, G., Rumbaugh, J., Rumbaugh, J., and Booch, G. (1999). *The unified software development process*, volume 1. Addison-wesley Reading.
- [Jania et al., 1996] Jania, J., Hagen, J. O., and Björnsson, H. (1996). *Mass balance of Arctic glaciers*. University of Silesia, Faculty of Earth Sciences.
- [Jaradat et al., 2013] Jaradat, W., Dearle, A., and Barker, A. (2013). An architecture for decentralised orchestration of web service workflows. In *Web Services (ICWS), 2013 IEEE 20th International Conference on*, pages 603–604.
- [Jonquet, 2006] Jonquet, C. (2006). *Dynamic Service Generation : Agent interactions for service exchange on the Grid*. PhD thesis, Université Montpellier II.
- [Jähne et al., 1999] Jähne, B., Haubecker, H., and Geibler, P. (1999). *Handbook of Computer Vision and Applications*, volume 1. Academic Press.
- [Kaplan and New, 2006] Kaplan, J. O. and New, M. (2006). Arctic climate change with a 2°C global warming : Timing, climate patterns and vegetation change. *Climatic Change*, 79 :213–241.
- [Karande et al., 2011] Karande, A., Karande, M., and B. B., M. (2011). Choreography and orchestration using business process execution language for soa with web services. *International Journal of Computer Science Issues*, 8.
- [Kergomard, 1982] Kergomard, C. (1982). *Recherches sur les climats océaniques du secteur arctique européen : Températures, circulation atmosphérique et glaces marines*. PhD thesis, Université des Sciences et Techniques de Lille.
- [Kergomard, 2007] Kergomard, C. (2007). L’arctique face au changement climatique. In *Annales de géographie*, number 1, pages 3–22. Armand Colin.
- [Klusch and Gerber, 2005] Klusch, M. and Gerber, A. (2005). Semantic web service composition planning with owls-xplan. In *In Proceedings of the 1st Int. AAAI Fall Symposium on Agents and the Semantic Web*, pages 55–62.
- [Kon et al., 2002] Kon, F., Costa, F., Blair, G., and Campbell, R. H. (2002). The case for reflective middleware. *Communications of the ACM*, 45(6) :33–38.
- [Kopecký et al., 2007] Kopecký, J., Vitvar, T., Bournez, C., and Farrell, J. (2007). Sawsdl : Semantic annotations for wsdl and xml schema. *IEEE Internet Computing*, 11(6) :60–67.
- [Kumar et al., 1994] Kumar, V., Grama, A., Gupta, A., and Karypis, G. (1994). *Introduction to parallel computing : design and analysis of algorithms*, volume 400. Benjamin/Cummings Redwood City, CA.
- [KVM, 2008] KVM (2008). Kernal Based Virtual Machine. [www.linux-kvm.org/page/Main\\_Page](http://www.linux-kvm.org/page/Main_Page).
- [Ladurie, 1959] Ladurie, E. L. R. (1959). Histoire et climat. In *Annales. Economies, Sociétés, Civilisations*, volume 14, pages 3–34. Persée-Portail des revues scientifiques en SHS.
- [Laffly et al., 2012a] Laffly, D., Bernard, E., Friedt, J.-M., Martin, G., Marlin, C., and Griselin, M. (2012a). Snow cover monitoring using combined formosat satellite imaging, oblique view ground-based pictures and snow drills. In *11th International Circumpolar Remote Sensing Symposium*.
- [Laffly et al., 2012b] Laffly, D., Bernard, E., Friedt, J.-M., Martin, G., Tolle, F., Marlin, C., and Griselin, M. (2012b). High temporal resolution monitoring of snow cover using oblique view grounded-based pictures. *Polar Record*, 48 :11–16.
- [Lajmi et al., 2007] Lajmi, S., Ghedira, C., and Ghedira, K. (2007). Une méthode d’apprentissage pour la composition de services web. In *Atelier Intelligence Artificielle et Web Intelligence (IAWI’07)*.

- [Lakshman and Malik, 2010] Lakshman, A. and Malik, P. (2010). Cassandra : a decentralized structured storage system. *ACM SIGOPS Operating Systems Review*, 44(2) :35–40.
- [Lämmermann, 2002] Lämmermann, S. (2002). *Runtime Service Composition via Logic-Based Program Synthesis*. PhD thesis, KTH, Superseded Departments, Microelectronics and Information Technology, IMIT.
- [Lamport, 1994] Lamport, L. (1994). *Latex*. Addison-Wesley.
- [Lapointe, 1993] Lapointe, J. (1993). L’approche systémique et la technologie de l’éducation. *Éducatechnologiques*, 1 :1–18.
- [Lecué, 2008] Lecué, F. (2008). *Composition de Services Web : Une Approche basée Liens Sémantiques*. Informatique, Ecole Nationale Supérieure des Mines de Saint-Etienne.
- [Legrand Contes, 2011] Legrand Contes, V. (2011). *Une approche à composant pour l’orchestration de services à large échelle*. PhD thesis, Université de Nice - Sophia Antipolis.
- [Li et al., 2010] Li, H., Spence, C., Armstrong, R., Godfrey, R., Schneider, R., Smith, J., and White, R. (2010). Intel cloud computing taxonomy and ecosystem analysis. Technical report, IT-Intel Brief (Cloud Computing).
- [Lin et al., 2009] Lin, G., Fu, D., Zhu, J., and Dasmalchi, G. (2009). Cloud computing : It as a service. *IT Professional Magazine*, 11(2) :10.
- [Lin, 2012] Lin, Y. (2012). *Méthodologie et composants pour la mise en oeuvre de workflows scientifiques*. PhD thesis, Université de Montpellier II.
- [Lliboutry, 1965] Lliboutry, L. (1965). *Traité de glaciologie*, volume 1. Masson.
- [Lohmann, 2007] Lohmann, N. (2007). A feature-complete petri net semantics for ws-bpel 2.0. In *International Workshop on Web Services and Formal Methods*, pages 77–91. Springer.
- [Lucchi and Mazzara, 2007] Lucchi, R. and Mazzara, M. (2007). A pi-calculus based semantics for ws-bpel. *The Journal of Logic and Algebraic Programming*, 70(1) :96–118.
- [Magnuson et al., 2000] Magnuson, J. J., Robertson, D. M., Benson, B. J., Wynne, R. H., Livingstone, D. M., Arai, T., Assel, R. A., Barry, R. G., Card, V., Kuusisto, E., et al. (2000). Historical trends in lake and river ice cover in the northern hemisphere. *Science*, 289(5485) :1743–1746.
- [Mann, 2002] Mann, M. E. (2002). Little ice age. *Encyclopedia of global environmental change*, 1 :504–509.
- [Marlin and Griselin, 2011] Marlin, C. and Griselin, M. (2011). Climate recent evolution observed in svalbard (air temperature and precipitation) : a focus on the 40 years in ny alesund (79° n). In *International Conference Polar Worlds*.
- [Marlin et al., 2009] Marlin, C., Griselin, M., Bernard, E., Delangle, E., Laffly, D., and Friedt, J.-M. (2009). Impact of climate conditions on the hydrological response of a polar glacier system-austrelovénbreen-svalbard. In *MOCA 09, Our warming Planet, Joint assembly, symposium J15 High Latitude Terrestrial Processes : Hydrology and Interactions with the Atmosphere*.
- [Marlin et al., 2008] Marlin, C., Griselin, M., Laffly, D., Grisolle, F., Noret, A., Friedt, J., and Martin, G. (2008). Impact of climate upon the hydrology of a small polar glacier. In *Hydrology in the Arctic Climate*.
- [Marlin et al., 2013] Marlin, C., Quenet, M., Griselin, M., Grenier, C., Roux, N., Tolle, F., Bernard, E., Friedt, J.-M., and Saintenoy, A. (2013). Relations entre eaux souterraines, cours d’eau et pergélisol au spitsberg occidental : exemple du bassin versant du glacier austre lovén (79° n, svalbard). In *Colloque Arctique, les grands enjeux scientifiques*.

- [Marroquin and Giroi, 1993] Marroquin, J. L. and Giroi, F. (1993). Some extensions of the k-means algorithm for image segmentation and pattern classification.
- [Matskin et al., 2005] Matskin, M., Kúngas, P., Rao, J., Sampson, J., Petersen, S. A., Link, I., and Back, J. (2005). Enabling web services composition with software agents. In *IMSA*, pages 93–98.
- [McDermott et al., 1998] McDermott, D., Ghallab, M., Howe, A., Knoblock, C., Ram, A., Veloso, M., Weld, D., and Wilkins, D. (1998). Pddl-the planning domain definition language.
- [McDermott, 2002] McDermott, D. V. (2002). Estimated-regression planning for interactions with web services. In *AIPS*, volume 2, pages 204–211.
- [McGuinness et al., 2012] McGuinness, D. L., Van Harmelen, F., et al. (2012). Owl web ontology language. W3C Recommendation. <https://www.w3.org/TR/ws-chor-reqs/>.
- [McIlraith and Son, 2002] McIlraith, S. and Son, T. C. (2002). Adapting golog for composition of semantic web services. *KR*, 2 :482–493.
- [McIlraith et al., 2001] McIlraith, S. A., Son, T. C., and Zeng, H. (2001). Semantic web services. *IEEE intelligent systems*, 16(2) :46–53.
- [Medjahed et al., 2003] Medjahed, B., Bouguettaya, A., and Elmagarmid, A. K. (2003). Composing web services on the semantic web. *The VLDB Journal – The International Journal on Very Large Data Bases*, 12(4) :333–351.
- [Meentemeyer, 1989] Meentemeyer, V. (1989). Geographical perspectives of space, time, and scale. *Landscape ecology*, 3(3-4) :163–173.
- [Meier et al., 2003] Meier, M. F., Dyurgerov, M. B., and McCabe, G. J. (2003). The health of glaciers : Recent changes in glacier regime. *Climatic Change*, 59(1-2) :123–135.
- [Mell and Grance, 2011] Mell, P. and Grance, T. (2011). The nist definition of cloud computing. NIST Special Publication 800-145.
- [Mercier, 1997] Mercier, D. (1997). L’impact du ruissellement sur les moraines latérales du glacier du roi. *Norvís*, 175(1) :549–566.
- [Mercier, 2000] Mercier, D. (2000). Du glaciaire au paraglaciaire la métamorphose des paysages polaires au svalbard/from glacial to paraglacial : the metamorphosis of polar landscapes in svalbard. In *Annales de Géographie*, pages 580–596. JSTOR.
- [Mercier, 2007] Mercier, D. (2007). Le paraglaciaire : évolution d’un concept. *Du continent au bassin versant. Théories et pratiques en géographie physique. Hommage au Professeur Alain Godard*, page 592.
- [Miller, 2013] Miller, J. J. (2013). Graph database applications and concepts with neo4j. In *Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA*, volume 2324.
- [MongoDB, 2016] MongoDB (2016). The mongodb manual. Technical report, MongoDB Inc. <https://docs.mongodb.com/manual/>.
- [Moniruzzaman and Hossain, 2013] Moniruzzaman, A. B. M. and Hossain, S. A. (2013). Nosql database : New era of databases for big data analytics - classification, characteristics and comparison. *International Journal of Database Theory and Application*, 6.
- [Moreau, 2005] Moreau, M. (2005). *Dynamique des paysages végétaux depuis la fin du petit âge glaciaire au Spitsberg (79° N)*. PhD thesis, Université Blaise Pascal - Clermont-Ferrand II.

- [Moreau et al., 2009] Moreau, M., Laffly, D., and Brossard, T. (2009). Recent spatial development of Svalbard strandflat vegetation over 31 years. *Polar Research*, 28 :364–375.
- [Moreau et al., 2005] Moreau, M., Laffly, D., Joly, D., and Brossard, T. (2005). Analysis of plant colonization on an arctic moraine since the end of the little ice age using remotely sensed data and a bayesian approach. *Remote Sensing of Environment*, pages 244–253.
- [Müller et al., 2006] Müller, I., Kowalczyk, R., and Braun, P. (2006). Towards agent-based coalition formation for service composition. In *Proceedings of the IEEE/WIC/ACM international conference on Intelligent Agent Technology*, pages 73–80. IEEE Computer Society.
- [Murray, 2009] Murray, P. (2009). Enterprise grade cloud computing. In *Proceedings of the Third Workshop on Dependable Distributed Data Management*, pages 1–1. ACM.
- [Narayanan and McIlraith, 2002] Narayanan, S. and McIlraith, S. A. (2002). Simulation, verification and automated composition of web services. In *Proceedings of the 11th international conference on World Wide Web*, pages 77–88. ACM.
- [Nau et al., 1999] Nau, D., Cao, Y., Lotem, A., and Munoz-Avila, H. (1999). Shop : Simple hierarchical ordered planner. In *Proceedings of the 16th international joint conference on Artificial intelligence-Volume 2*, pages 968–973. Morgan Kaufmann Publishers Inc.
- [Nau et al., 2003] Nau, D., Ilghami, O., Kuter, U., Murdock, J. W., Wu, D., and Yaman, F. (2003). Shop2 : An htn planning system. *Journal of Artificial Intelligence Research*, 20 :379–404.
- [Numérique, 2012] Numérique, S. (2012). Livre blanc cloud computing "nouveaux modèles!".
- [Oerlemans, 2005] Oerlemans, J. (2005). Extracting a climate signal from 169 glacier records. *Science*, 308(5722) :675–677.
- [Oh et al., 2010] Oh, T. H., Lim, S., Choi, Y. B., Park, K.-R., Lee, H., and Choi, H. (2010). State of the art of network security perspectives in cloud computing. In *Security-Enriched Urban Computing and Smart Grid*, pages 629–637. Springer.
- [Overland et al., 2004] Overland, J. E., Spillane, M. C., Percival, D. B., Wang, M., and Mofjeld, H. O. (2004). Seasonal and regional variation of pan-arctic surface air temperature over the instrumental record. *Journal of Climate*, 17(17) :3263–3282.
- [Parkhill, 1966] Parkhill, D. F. (1966). *The Challenge of the Computer Utility*. Addison-Wesley Publishing Company.
- [Pattison, 1964] Pattison, W. D. (1964). The four traditions of geography. *Journal of Geography*, 63(5) :211–216.
- [Péguy and Brossard, 1993] Péguy, C.-P. and Brossard, T. (1993). Chronique : Trente ans de présence française au spitsberg (1963-1993)/thirty years of french presence in spitsberg (1963-1993). *Revue de géographie de Lyon*, 68(2-3) :203–205.
- [Peltz, 2003] Peltz, C. (2003). Web services orchestration and choreography. *Computer*, 36(10) :46–52.
- [Perlovsky and McManus, 1991] Perlovsky, L. I. and McManus, M. M. (1991). Maximum likelihood neural networks for sensor fusion and adaptive classification. *Neural Networks*, 4(1) :89–102.
- [Pordes et al., 2007] Pordes, R., Petravick, D., Kramer, B., Olson, D., Livny, M., Roy, A., Avery, P., Blackburn, K., Wenaus, T., Frank, W., et al. (2007). The open science grid. In *Journal of Physics : Conference Series*, volume 78, page 012057. IOP Publishing.

- [Preist et al., 2003] Preist, C., Bartolini, C., and Byde, A. (2003). Agent-based service composition through simultaneous negotiation in forward and reverse auctions. In *Proceedings of the 4th ACM conference on Electronic commerce*, pages 55–63. ACM.
- [Preist et al., 2001] Preist, C., Byde, A., Bartolini, C., and Piccinelli, G. (2001). Towards agent-based service composition through negotiation in multiple auctions. *AISB Journal*, 1(1) :109–124.
- [Qian et al., 2009] Qian, L., Luo, Z., Du, Y., and Guo, L. (2009). Cloud computing : an overview. In *IEEE International Conference on Cloud Computing*, pages 626–631. Springer.
- [Quenet, 2014] Quenet, M. (2014). *Rôle des eaux souterraines dans l’hydrologie d’un bassin versant glaciaire sous condition de pergélisol continu au Spitsberg (Austre Lovénbreen, 79° N) : approches hydrodynamiques et géochimiques*. PhD thesis, Paris 11.
- [Ramakrishnan and Gehrke, 2000] Ramakrishnan, R. and Gehrke, J. (2000). *Database management systems*. McGraw-Hill.
- [Ranisavljević et al., 2013] Ranisavljević, E., Devin, F., Laffly, D., and Nir, Y. L. (2013). Semantic orchestration of image processing services for environmental analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 83 :184–192.
- [Ranisavljević et al., 2014a] Ranisavljević, E., Devin, F., Laffly, D., and Nir, Y. L. (2014a). A dynamic and generic cloud computing model for glaciological image processing. *International Journal of Applied Earth Observation and Geoinformation*, 27 :109–115.
- [Ranisavljević et al., 2014b] Ranisavljević, E., Devin, F., Nir, Y. L., and Laffly, D. (2014b). Architecture comparison of an in-situ data processing application applied to glacier mass balance analyse. In *13th International Circumpolar Remote Sensing Symposium*.
- [Rao, 2004] Rao, J. (2004). *Semantic Web Service Composition via Logic-based Program Synthesis*. PhD thesis, Norwegian University of Science and Technology, Faculty of Information Technology, Mathematics and Electrical Engineering, Department of Computer and Information Science.
- [Rao et al., 2004] Rao, J., Kungas, P., and Matskin, M. (2004). Logic-based web services composition : from service description to process model. In *Web Services, 2004. Proceedings. IEEE International Conference on*, pages 446–453.
- [Reed, 2003] Reed, D. A. (2003). Grids, the teragrid and beyond. *Computer*, 36(1) :62–68.
- [Reynaud et al., 1986] Reynaud, L., Vallon, M., and Letréguilly, A. (1986). Mass-balance measurements : problems and two new methods of determining variations. *Journal of Glaciology*, 32(112) :446–454.
- [Reynaud and Vincent, 2002] Reynaud, L. and Vincent, C. (2002). Histoire des fluctuations des glaciers en remontant le petit age de glace. *Houille blanche*, 8(8) :16–19.
- [Rimal et al., 2009] Rimal, B. P., Choi, E., and Lumb, I. (2009). A taxonomy and survey of cloud computing systems. *INC, IMS and IDC*, pages 44–51.
- [Ripley, 2007] Ripley, B. D. (2007). *Pattern recognition and neural networks*. Cambridge university press.
- [Rolstad et al., 2000] Rolstad, C., Whillans, I., Hagen, J. O., and Isaksson, E. (2000). Large-scale force budget of an outlet glacier : Jutulstraumen, dronning maud land, east antarctica. *Annals of Glaciology*, 30(1) :35–41.
- [Romanovsky et al., 2010] Romanovsky, V. E., Smith, S. L., and Christiansen, H. H. (2010). Permafrost thermal state in the polar northern hemisphere during the international polar year 2007–2009 : a synthesis. *Permafrost and Periglacial Processes*, 21(2) :106–116.

- [Sadashiv and Kumar, 2011] Sadashiv, N. and Kumar, S. D. (2011). Cluster, grid and cloud computing : A detailed comparison. In *Computer Science & Education (ICCSE), 2011 6th International Conference on*, pages 477–482. IEEE.
- [Saintenoy et al., 2013] Saintenoy, A., Friedt, J.-M., Booth, A. D., Tolle, F., Bernard, E., Laffly, D., Marlin, C., and Griselin, M. (2013). Deriving ice thickness, glacier volume and bedrock morphology of the austre lov\`enbreen (svalbard) using ground-penetrating radar. *arXiv preprint arXiv :1306.2539*.
- [Saintenoy et al., 2011] Saintenoy, A., Friedt, J.-M., Tolle, F., Bernard, E., Laffly, D., Marlin, C., and Griselin, M. (2011). High density coverage investigation of the austre lovénbreen (svalbard) using ground penetrating radar. In *Advanced Ground Penetrating Radar (IWAGPR), 2011 6th International Workshop on*, pages 1–4. IEEE.
- [Salas et al., 2010] Salas, D. et al. (2010). Retrait de la banquise arctique : retour sur les dernières décennies. *Météorologie*, (71) :30–35.
- [Schlimmer, 2002] Schlimmer, J. C. (2002). Web services description requirements. W3C Working Draft. <http://www.w3.org/TR/ws-desc-reqs/>.
- [Serreze et al., 2000] Serreze, M., Walsh, J., Chapin Iii, F., Osterkamp, T., Dyurgerov, M., Romanovsky, V., Oechel, W., Morison, J., Zhang, T., and Barry, R. (2000). Observational evidence of recent change in the northern high-latitude environment. *Climatic Change*, 46(1-2) :159–207.
- [Serreze and Francis, 2006] Serreze, M. C. and Francis, J. A. (2006). The arctic amplification debate. *Climatic change*, 76(3-4) :241–264.
- [Shimizu and Fuse, 2003] Shimizu, E. and Fuse, T. (2003). Rubber-sheeting of historical maps in gis and its application to landscape visualization of old-time cities : focusing on tokyo of the past. In *Proceedings of the 8th international conference on computers in urban planning and urban management*, volume 11, page 3.
- [Sirin et al., 2004] Sirin, E., Parsia, B., Wu, D., Hendler, J., and Nau, D. (2004). Htn planning for web service composition using shop2. *Web Semantics : Science, Services and Agents on the World Wide Web*, 1(4) :377–396.
- [Soruco et al., 2011] Soruco, A., Thibert, E., Vincent, C., Blanc, R., and Héno, R. (2011). Measurement of avalanche front velocity from high-speed terrestrial digital photogrammetry. In *Geophys. Res. Abstr*, volume 13.
- [Srinivasan and Treadwell, 2005] Srinivasan, L. and Treadwell, J. (2005). An overview of service-oriented architecture, web services and grid computing. *HP Software Global Business Unit*, 2 :1–13.
- [Stroeve et al., 2007] Stroeve, J., Holland, M. M., Meier, W., Scambos, T., and Serreze, M. (2007). Arctic sea ice decline : Faster than forecast. *Geophysical research letters*, 34(9).
- [Sumbaly et al., 2012] Sumbaly, R., Kreps, J., Gao, L., Feinberg, A., Soman, C., and Shah, S. (2012). Serving large-scale batch computed data with project voldemort. In *Proceedings of the 10th USENIX conference on File and Storage Technologies*, pages 18–18. USENIX Association.
- [Sund and Eiken, 2010] Sund, M. and Eiken, T. (2010). Recent surges on blomstrandbreen, comfortlessbreen and nathorstbreen, svalbard. *Journal of Glaciology*, 56(195) :182.
- [Sycara et al., 2004] Sycara, K., Paolucci, M., Soudry, J., and Srinivasan, N. (2004). Dynamic discovery and coordination of agent-based semantic web services. *IEEE Internet computing*, 8(3) :66–73.



- [Tanenbaum and Van Steen, 2007] Tanenbaum, A. S. and Van Steen, M. (2007). *Distributed systems*. Prentice-Hall.
- [van der Aalst and ter Hofstede, 2005] van der Aalst, W. and ter Hofstede, A. (2005). Yawl : yet another workflow language. *Information Systems*, 30(4) :245 – 275.
- [Vaughan et al., 2013] Vaughan, D. G., Comiso, J. C., Allison, I., Carrasco, J., Kaser, G., Kwok, R., Mote, P., Murray, T., Paul, F., Ren, J., et al. (2013). Observations : cryosphere. *Climate change*, pages 317–382.
- [VMware, 1998] VMware (1998). VMware EVO SDDC. [www.vmware.com/products/evosddc](http://www.vmware.com/products/evosddc).
- [Wang et al., 2010] Wang, L., Von Laszewski, G., Younge, A., He, X., Kunze, M., Tao, J., and Fu, C. (2010). Cloud computing : a perspective study. *New Generation Computing*, 28(2) :137–146.
- [Weglarz, 2004] Weglarz, G. (2004). Two worlds data-unstructured and structured. *DM REVIEW*, 14 :19–23.
- [Weiss, 2007] Weiss, A. (2007). Computing in the clouds. *Computing*, 11 :16–25.
- [WGMS, 2008] WGMS (2008). Global glacier changes : facts and figures. Technical report, UNEP.
- [White et al., 2007] White, D., Hinzman, L., Alessa, L., Cassano, J., Chambers, M., Falkner, K., Francis, J., Gutowski, W. J., Holland, M., Holmes, R. M., et al. (2007). The arctic freshwater system : Changes and impacts. *Journal of Geophysical Research : Biogeosciences*, 112(G4).
- [Wolfgang, 1994] Wolfgang, P. (1994). *Design patterns for object-oriented software development*. Reading, Mass. : Addison-Wesley.
- [Wolkowicz et al., 2012] Wolkowicz, H., Saigal, R., and Vandenberghe, L. (2012). *Handbook of semidefinite programming : theory, algorithms, and applications*, volume 27. Springer Science & Business Media.
- [Wooldridge and Jennings, 1995] Wooldridge, M. and Jennings, N. R. (1995). Intelligent agents : Theory and practice. *The knowledge engineering review*, 10(02) :115–152.
- [Wu et al., 2003] Wu, D., Sirin, E., Hendler, J., Nau, D., and Parsia, B. (2003). Automatic web services composition using shop2. Workshop on Planning for Web Services.
- [Xen, 2003] Xen (2003). Xen Source Inc. [www.xen-source.com](http://www.xen-source.com).
- [Yang and Papazoglou, 2004] Yang, J. and Papazoglou, M. P. (2004). Service components for managing the life-cycle of service compositions. *Information Systems*, 29(2) :97–125.
- [Zaslavsky et al., 2013] Zaslavsky, A., Perera, C., and Georgakopoulos, D. (2013). Sensing as a service and big data. In *Proceedings of the International Conference on Advances in Cloud Computing*.
- [Zhang et al., 2010] Zhang, Q., Cheng, L., and Boutaba, R. (2010). Cloud computing : state-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1(1) :7–18.
- [Zhou et al., 2010] Zhou, M., Zhang, R., Zeng, D., and Qian, W. (2010). Services in the cloud computing era : A survey. In *Universal Communication Symposium (IUCS), 2010 4th International*, pages 40–46. IEEE.
- [Zollmann, 2012] Zollmann, J. (2012). Nosql databases. Retrieved from *Software Engineering Research Group* : <http://www.webcitation.org/6hA9zoqRd>.
- [Zryd, 2001] Zryd, A. (2001). *Les glaciers*, volume 4. Editions Saint-Augustin.
- [Zube et al., 1982] Zube, E. H., Sell, J. L., and Taylor, J. G. (1982). Landscape perception : research, application and theory. *Landscape planning*, 9(1) :1–33.

# Liste des acronymes

ACID	<i>Atomicity, Consistency, Isolation, Durability.</i> 54, 56
AUV	<i>Unmanned Aerial Vehicles.</i> 11
BDD	Base De Données. 53, 54, 56, 58, 59, 121, 122, 124
BPEL	<i>Business Process Execution Language.</i> 80, 81, 85, 103, 113, 121
BSON	<i>Binary JSON.</i> 59
CAP	<i>Consistency, Availability, Partition tolerance.</i> 56, 59
CPU	<i>Central Processing Unit.</i> 122–125, 181
GCP	points de contrôle ou <i>Ground Control Points.</i> 67, 72
HSF	Hydro-Sensor-FLOWS. 12, 13, 28, 36, 37, 39– 42, 44, 48, 51, 60–62, 64, 65, 71, 74, 85, 88, 119, 153, 180, 185
HTN	<i>Hierarchical Task Network.</i> 84
HTTP	<i>Hypertext Transfer Protocol.</i> 76
IaaS	<i>Infrastructure as a Service.</i> 33, 34
IPCC	<i>Intergovernmental Panel on Climate Change.</i> 22
JSON	<i>JavaScript Object Notation.</i> 57, 59, 180
NoSQL	<i>Not Only SQL.</i> 55, 56, 58, 59, 121
OWL	<i>Web Ontology Language.</i> 83
PaaS	<i>Platform as a service.</i> 33, 34
PDDL	<i>Planning Domain Definition Language.</i> 83, 84

RAM	<i>Random Access Memory.</i> 122, 124, 125, 181
RDF	<i>Resource Description Framework.</i> 83
RDF-S	<i>RDF Schema.</i> 83
SaaS	<i>Software as a Service.</i> 33, 34
SAWSDL	<i>Semantic Annotation for WSDL.</i> 83, 103
SGBD	<i>Système de Gestion de Base de Données.</i> 53–55
SHOP	<i>Simple Hierarchical Ordered Planner.</i> 84
SIG	<i>Système d'Information Géographique.</i> 12
SOA	<i>Service-Oriented Architecture.</i> 76–79
SOAP	<i>Simple Object Access Protocol.</i> 76, 103
SPARQL	<i>SPARQL Protocol and RDF Query Language.</i> 113
SQL	<i>Structured Query Language.</i> 54
SVM	<i>Support Vector Machine.</i> 69
UDDI	<i>Universal Description Discovery and Integration.</i> 77
URI	<i>Uniform Resource Identifier.</i> 75, 83
URL	<i>Uniform Resource Locator.</i> 76
W3C	<i>World Wide Web Consortium.</i> 75, 80
WS	<i>Web Service.</i> 75–77, 80, 82, 84, 85, 88, 94, 102, 103, 181
WSDL	<i>Web Service Description Language.</i> 76, 77, 83, 103
XML	<i>Extensible Markup Language.</i> 75, 76, 80, 83
XSD	<i>XML Schema Definition.</i> 76

# Glossaire

élasticité	est l'évolution de la consommation des ressources en fonction de la demande, autant par rapport au nombre de clients que par rapport au nombre de ressources de stockage ou de calcul nécessaire. 35, 56
bande passante	indique la capacité d'un réseau à transmettre des informations. 122, 123
bassin versant	est une portion de territoire dont l'ensemble des eaux convergent vers un même point de sortie appelé exutoire : cours d'eau, lac, mer, océan, etc.. 38, 42, 45–47, 51, 119, 128, 178, 184
BDD	correspond à une collection d'information, de données et de liens entre ces données, qui existent sur une longue période de temps, souvent plusieurs années. 53
classification non supervisée	est une méthode d'apprentissage automatique. Elle fonctionne en divisant un groupe hétérogène de données en sous-groupes, de manière à ce que les données considérées comme les plus similaires soient associées au sein d'un groupe homogène et qu'à l'opposé, les données considérées comme différentes se retrouvent dans d'autres groupes distincts. 69
classification supervisée	est une technique d'apprentissage automatique où l'on cherche à produire automatiquement des règles à partir d'une base de données d'apprentissage contenant des "exemples" (en général des cas déjà traités et validés). 69
CPU	aussi appelé processeur, est un composant présent dans de nombreux dispositifs électroniques qui exécute les instructions machine des programmes informatiques. 122

cryosphère	est un terme désignant collectivement les portions de la surface de la Terre où l'eau est présente à l'état solide. Elle inclut les banquises, les lacs et rivières gelés, les régions couvertes de neige, les glaciers, les inlandsis et les sols gelés, de façon temporaire ou permanente (pergélisol). 23
endpoint	est l'URL par laquelle un client peut accéder au service. 76
EXIF	est une spécification de format de fichier pour les images utilisées par les appareils photographiques numériques pour stocker des métadonnées sur la photo acquise comme la date, l'heure, les réglages de l'appareil, etc. 65
map	aussi appelé dictionnaire ou tableau d'associatif, est un type de données associant à un ensemble de clefs un ensemble correspondant de valeurs. Chaque clef, unique dans l'ensemble des clefs, est associée à une valeur. 56
moraine	est formée par le matériel qui provient de l'érosion et du dépôt liés au glacier. Le matériel est très hétérogène : il se compose de particules fines comme la farine glaciaire, de roches et jusqu'à de très gros blocs dit erratiques. 25
ontologie	est un réseau sémantique qui regroupe un ensemble de concepts décrivant complètement un domaine. Ces concepts sont liés les uns aux autres par des relations taxinomiques (hiérarchisation des concepts) d'une part, et sémantiques d'autre part. 85
périglaciale	est un phénomène lié au gel de l'eau. 24
paraglaciale	est un phénomène lié à la fonte de la glace. 24
pergélisol	désigne la partie d'un cryosol gelé en permanence, au moins pendant deux ans, et de ce fait imperméable. 23, 38
RAM	aussi appelé mémoire vive, est la mémoire informatique dans laquelle sont rangées les informations (données, instructions, etc) en traitement par un appareil (par exemple, un ordinateur). 122

sémantique	permet de rajouter du sens aux différentes informations présentes. 83
scalabilité	désigne la capacité d'un produit à s'adapter à un changement d'ordre de grandeur de la demande (montée en charge), en particulier sa capacité à maintenir ses fonctionnalités et ses performances en cas de forte demande. 35, 120, 121
schéma	est, en base de données, la description de l'organisation des données. Il se trouve à l'intérieur de la base de données, et renseigne sur les caractéristiques de chaque type de donnée et les relations entre les différentes données qui se trouvent dans la base de données. 53-55
transaction	est une unité de travail, encapsulant généralement un certain nombre d'opérations sur une base de données (par exemple, la lecture, l'écriture ou la suppression de données). 54

# Table des figures

1	Schéma de l'organisation de la première partie de la thèse . . . . .	3
1.1	Évolution du glacier d'Argentière en un siècle . . . . .	10
1.2	Emplacement des changements significatifs relevés dans les séries de données sur les systèmes physiques et les systèmes biologiques et variations de la température de l'air en surface pendant la période 1970-2004 . . . . .	11
1.3	Les différentes composants de la cryosphère . . . . .	12
1.4	La limite de la région arctique . . . . .	13
1.5	Longyearbyen, Spitsberg, 2013 . . . . .	14
1.6	Schéma synthétique du fonctionnement d'un glacier . . . . .	15
1.7	Formes et processus paraglaciers . . . . .	16
1.8	Localisation de l'île du Spitsberg. . . . .	17
1.9	Carte du Spitsberg du XVII <sup>ème</sup> siècle . . . . .	17
1.10	Base française Corbel, proche de Ny-Ålesund, en août 2012. . . . .	18
2.1	<i>Grids et Clouds</i> . . . . .	21
2.2	L'architecture du <i>cloud computing</i> . . . . .	22
2.3	Différents types de modèle de services dans le <i>cloud computing</i> . . . . .	23
2.4	Définition de la stratégie de déploiement . . . . .	25
3.1	Localisation du Spitsberg et du Lovén Est . . . . .	27
3.2	Le glacier du Lovén Est en retrait depuis 1936 . . . . .	28
3.3	Carte et photos des deux exutoires du bassin versant du Lovén Est . . . . .	28
3.4	Réseau des appareils de mesures sur le Lovén Est . . . . .	29
3.5	Températures annuelles de l'air à Ny Ålesund (1969-2010) . . . . .	30
3.6	Précipitations annuelles à Ny Ålesund (1969-2010) . . . . .	30
3.7	Détermination de l'état thermique du glacier . . . . .	31
3.8	Roses des vents sur le glacier Lovén Est . . . . .	32
3.9	Reconstitution du glacier à partir des photos . . . . .	32
3.10	Évolution de la fonte en trois ans . . . . .	33
3.11	Schéma complet de l'évolution de la couverture neigeuse couplé aux températures et aux précipitations . . . . .	33
3.12	Suivi des avalanches à partir des clichés des stations photo automatiques . . . . .	34
3.13	Hauteurs de neige et équivalents en eau du manteau neigeux à la fin de l'hiver . . . . .	35
3.14	Chronique du débit de la Goule et précipitations en 2009 . . . . .	36
3.15	Chronique du débit de la Goule et températures en 2009 . . . . .	36
3.16	Chronique du débit de la Diable et précipitations en 2009 . . . . .	37

3.17	Chronique du débit de la Diable et températures en 2009 . . . . .	37
3.18	Les deux exutoires du Lovén Est . . . . .	38
3.19	Bilans de masse annuels du Lovén Est . . . . .	39
3.20	Position du front du glacier Lovén Est depuis 2007 . . . . .	39
3.21	Trois radargrammes illustrant les grands axes du glacier et un exemple de la langue . . . . .	40
4.1	Exemple de données représentées dans un stockage <i>clé-valeur</i> . . . . .	47
4.2	Exemple de données représentées dans un stockage <i>document</i> . . . . .	47
4.3	Exemple de données représentées dans un stockage <i>orienté colonnes</i> . . . . .	48
4.4	Exemple de données représentées dans un stockage <i>graphe</i> . . . . .	48
5.1	Acquisitions des images FORMOSAT au cours du programme HSF . . . . .	52
5.2	Deux clichés satellite FORMOSAT du glacier Lovén Est, acquises en 6 semaines . . . . .	52
5.3	La vue de chacune des 6 caméras, ainsi que leur position dans le glacier . . . . .	53
5.4	La mosaïque construite avec les photos référentielles . . . . .	53
5.5	La mosaïque construite avec les photos référentielles . . . . .	54
5.6	Traitement de projection . . . . .	56
5.7	Traitement de mosaïque . . . . .	58
5.8	Traitement de classification . . . . .	59
5.9	Différents types de perturbations météorologiques . . . . .	60
5.10	Caméra dans sa boîte avec les panneaux solaire . . . . .	61
5.11	Ombres portées pour une caméra donnée . . . . .	62
5.12	Conséquences du climat extrême sur les stations photos . . . . .	63
5.13	Décalage du champ de vision d'une caméra entre 2007 et 2008 . . . . .	63
5.14	Image après recadrage. La zone rougeâtre est perdue. . . . .	64
6.1	Relation entre les technologies utilisées pour les WS . . . . .	66
6.2	Orchestration vs chorégraphie . . . . .	69
6.3	Structure d'un agent offrant un service . . . . .	72
10.1	Modèle d'une architecture de traitement d'images dans le <i>cloud computing</i> . . . . .	111
10.2	Mesure du temps d'exécution pour les 3 architectures . . . . .	113
10.3	Mesure de la bande passante utilisée pour les 3 architectures . . . . .	113
10.4	Mesure de l'utilisation du CPU pour les 3 architectures . . . . .	114
10.5	Mesure de l'utilisation de la RAM pour les 3 architectures . . . . .	114



# Table des matières

<b>Remerciements</b>	<b>iii</b>
<b>Introduction générale</b>	<b>1</b>
Contexte général . . . . .	1
Motivations et objectifs . . . . .	1
La thèse dans ses grandes lignes . . . . .	2
<b>I GÉOGRAPHIE ET <i>Cloud computing</i>, UN NOUVEL HORIZON</b>	<b>5</b>
<b>1 Approche géographique de l'évolution des paysages polaires dans le cadre du changement climatique contemporain</b>	<b>9</b>
1.1 Le changement climatique global contemporain . . . . .	10
1.2 L'Arctique face aux changements climatiques . . . . .	12
1.3 Les glaciers au cœur du paysage polaire . . . . .	14
1.4 Les moraines, paysage du retrait glaciaire . . . . .	15
1.5 Le cas de la recherche au Spitsberg . . . . .	17
<b>2 Le cloud aux limites techniques de la géographie</b>	<b>19</b>
2.1 La naissance du <i>cloud computing</i> . . . . .	19
2.2 Le <i>cloud</i> , qu'est ce que c'est ? . . . . .	20
2.3 L'architecture du <i>cloud</i> . . . . .	22
2.4 Le <i>cloud</i> au service . . . . .	23
2.5 Les structures de déploiement . . . . .	24
<b>3 Le projet Hydro-Sensor-FLOWS, un glacier sous surveillance</b>	<b>27</b>
3.1 Le programme Hydro-Sensor-FLOWS . . . . .	28
3.2 Les données, les analyses . . . . .	29
3.2.1 Changement climatique récent . . . . .	30
3.2.2 Dynamique du manteau neigeux et suivi glaciologique du glacier Lovén Est . . . . .	31
3.2.3 Suivi hydrologique du bassin versant du Lovén Est . . . . .	36
3.2.4 La géomatique pour quantifier la réaction du glacier aux changements climatiques . . . . .	38
<b>4 La variété, l'hétérogénéité et la quantité des données : une base de données <i>cloud</i> adaptée</b>	<b>43</b>

4.1	Base de données structurées . . . . .	43
4.2	Variété et hétérogénéité de la géographie : la limite du relationnel . . . . .	44
4.3	Vers un modèle non structuré . . . . .	45
4.4	Bases de données NoSQL, une nouvelle ouverture . . . . .	46
4.5	MongoDB, une base de données adaptées . . . . .	49
<b>5</b>	<b>Analyse de l'évolution de la couverture neigeuse par photo <i>in situ</i></b>	<b>51</b>
5.1	La surveillance du manteau neigeux . . . . .	51
5.2	Le traitement des données <i>in situ</i> . . . . .	55
5.2.1	Projection . . . . .	56
5.2.2	Mosaïque . . . . .	57
5.2.3	Classification . . . . .	58
5.2.4	Nettoyage et utilisabilité . . . . .	60
5.2.5	Datation . . . . .	61
5.2.6	Recadrage . . . . .	62
5.3	Les limites des traitements . . . . .	64
<b>6</b>	<b>La composition de services de traitements géographiques</b>	<b>65</b>
6.1	Web Services . . . . .	65
6.2	La composition de services . . . . .	68
6.3	Méthodes de composition de services . . . . .	70
6.3.1	Composition de services via le business process . . . . .	70
6.3.2	Composition de services via la coordination d'agents . . . . .	71
6.3.3	Composition de services via la planification . . . . .	72
6.4	La composition pour la géographie . . . . .	75
<b>7</b>	<b>La géographie, laboratoire d'essai à grandeur nature pour le <i>cloud</i></b>	<b>77</b>
<b>II</b>	<b>L'ARCHITECTURE ET LE MODÈLE</b>	<b>79</b>
<b>8</b>	<b>Modèle d'application dans le <i>cloud</i></b>	<b>83</b>
8.1	Introduction partielle . . . . .	83
8.2	Article " <i>A dynamic and generic cloud computing model for environmental analysis using in situ data applied to glacier mass balance</i> " . . . . .	84
8.3	Conclusion partielle . . . . .	92
<b>9</b>	<b>L'orchestration des services</b>	<b>93</b>
9.1	Introduction partielle . . . . .	93
9.2	Article " <i>Semantic orchestration of image processing services for environmental analysis</i> " . . . . .	93
9.3	Conclusion partielle . . . . .	103
<b>III</b>	<b>RÉSULTATS</b>	<b>105</b>
<b>10</b>	<b>Comparaison des architectures</b>	<b>109</b>

10.1	Introduction : le projet Hydro-Sensor-FLOWS . . . . .	109
10.2	Modèle et Méthode . . . . .	110
10.3	Résultats et Discussion . . . . .	112
10.4	Conclusion . . . . .	115
<b>11</b>	<b>Réduction des dimensions des données</b>	<b>117</b>
11.1	Une masse de données géographiques grandissante . . . . .	117
11.2	Réduction de dimension . . . . .	118
11.2.1	La réduction de la taille d'une image . . . . .	118
11.2.2	La réduction de dimension des données réduites . . . . .	119
11.2.3	Position du problème . . . . .	119
11.3	Article draft : " <i>Wavelet Transform and Dimensionality Reduction methods for analyzing in-situ sensing images database</i> " . . . . .	120
11.4	Conclusion partielle . . . . .	141
	<b>Conclusion générale</b>	<b>142</b>
	Conclusion . . . . .	143
	Perspectives . . . . .	144
	<b>ANNEXE</b>	<b>145</b>
	<b>Bibliographie</b>	<b>149</b>
	<b>Liste des acronymes</b>	<b>163</b>
	<b>Glossaire</b>	<b>165</b>
	<b>Table des figures</b>	<b>168</b>
	<b>Table des matières</b>	<b>170</b>

**Titre** *Cloud computing* appliqué au traitement multimodal d'images *in situ* pour l'analyse des dynamiques environnementales

**Résumé** L'analyse des paysages, de ses dynamiques et ses processus environnementaux, nécessite d'acquérir régulièrement des données des sites, notamment pour le bilan glaciaire au Spitsberg et en haute montagne. À cause des mauvaises conditions climatiques communes aux latitudes polaires et à cause de leur coût, les images satellites journalières ne sont pas toujours accessibles. De ce fait, les événements rapides comme la fonte de la neige ou l'enneigement ne peuvent pas être étudiés à partir des données de télédétection à cause de leur fréquence trop faible. Nous avons complété les images satellites par un ensemble de stations photo automatiques et autonomes qui prennent 3 photos par jour. L'acquisition de ces photos génère une grande base de données d'images. Plusieurs traitements doivent être appliqués sur les photos afin d'extraire l'information souhaitée (modifications géométriques, gestion des perturbations atmosphériques, classification, etc). Seule l'informatique est à même de stocker et gérer toutes ces informations. Le cloud computing offre en tant que services des ressources informatiques (puissance de calcul, espace de stockage, applications, etc). Uniquement le stockage de la masse de données géographique pourrait être une raison d'utilisation du cloud computing. Mais en plus de son espace de stockage, le cloud offre une simplicité d'accès, une architecture scalable ainsi qu'une modularité dans les services disponibles. Dans le cadre de l'analyse des photos *in situ*, le cloud computing donne la possibilité de mettre en place un outil automatique afin de traiter l'ensemble des données malgré la variété des perturbations ainsi que le volume de données. À travers une décomposition du traitement d'images en plusieurs tâches, implémentées en tant que web services, la composition de ces services nous permet d'adapter le traitement aux conditions de chacune des données.

**Mots clés** Cloud computing ; services web ; composition de services ; NoSQL ; paysages ; *in situ sensing* ; dynamique environnementale ; géoscience

**Abstract** Analyzing landscape, its dynamics and environmental evolutions require regular data from the sites, specifically for glacier mass balanced in Spitsbergen and high mountain area. Due to poor weather conditions including common heavy cloud cover at polar latitudes, and because of its cost, daily satellite imaging is not always accessible. Besides, fast events like flood or blanket of snow is ignored by satellite based studies, since the slowest sampling rate is unable to observe it. We complement satellite imagery with a set of ground based autonomous automated digital cameras which take 3 pictures a day. These pictures form a huge database. Each picture needs many processing to extract the information (geometric modifications, atmospheric disturbances, classification, etc). Only computer science is able to store and manage all this information. Cloud computing, being more accessible in the last few years, offers as services IT resources (computing power, storage, applications, etc.). The storage of the huge geographical data could, in itself, be a reason to use cloud computing. But in addition to its storage space, cloud offers an easy way to access, a scalable architecture and a modularity in the services available. As part of the analysis of *in situ* images, cloud computing offers the possibility to set up an automated tool to process all the data despite the variety of disturbances and the data volume. Through decomposition of image processing in several tasks, implemented as web services, the composition of these services allows us to adapt the treatment to the conditions of each of the data.

**Keywords** Cloud computing ; web services ; services composition ; NoSQL ; landscapes ; *in situ sensing* ; environmental dynamics ; geoscience