

CURRICULUM VITAE

Identification et coordonnées

Nom **Guy Tremblay**
Statut Professeur associé/honoraire
Adresse Département d'informatique, UQAM
C.P. 8888, Succ. Centre-ville
Montréal, QC
H3C 3P8
Courriel `tremblay.guy@uqam.ca`

Formation universitaire

Ph.D., Computer Science, McGill University, 1994.
“*Parallel Implementation of Lazy Functional Languages using Abstract Demand Propagation*”
M. Math., Computer Science, University of Waterloo, 1984.
“*Strategies for Teaching Undergraduate Programming*”
B. Sc., Mathématiques (option informatique), UQAM, 1983.

Expérience professionnelle

- Professeur associé/honoraire, Dépt d'informatique, UQAM, 2020–
- Professeur (invité, substitut, régulier, titulaire), Dépt d'informatique, UQAM, 1985–2020
Sauf : Congé sans solde (thèse de doctorat) : 1992–1994
- Chercheur invité, Équipe Alpha, Dipartimento di Informatica, Università degli Studi di Torino, Turin, Italie, janvier–mai 2015 + juin 2016 + juillet 2017.
- Chercheur invité, Équipe D'OC, Laboratoire d'Informatique, Robotique et Micro-Électronique de Montpellier (LIRMM), Montpellier, France, janvier–juin 2007 + mai–juin 2008.
- Chercheur invité, Dept of EECS, University of Delaware, Newark, DE, janvier–juin 2000.
- Professeur invité, Institut Francophone d'Informatique, Hanoï, Vietnam, automne 1999.
- Professeur (substitut), Dépt de mathématiques et d'informatique, UQTR, Trois-Rivières, QC, 1985.
- Chargé de cours, Dépt de mathématiques et informatique, UQAM, été 1983.
- Analyste-programmeur, Recherche et développement, Les systèmes d'ordinateurs LOGO, Montréal, QC, 1982–1983.

Prix et reconnaissance

Prix Bâtitseur de la Faculté des sciences, volet études, 2022

Prix d'excellence en enseignement de l'UQAM, 2011

Prix d'excellence en enseignement de la Faculté des sciences de l'UQAM, 2010

Certificate of appreciation for contributing to the Guide to the Software Engineering Body of Knowledge as Knowledge Area Specialist in “Software Design”, IEEE Computer Society, 2001.

ENSEIGNEMENT

Domaines d'enseignement

- Méthodes formelles de modélisation et spécification.
- Programmation parallèle.
- Conception et analyse d'algorithmes (séquentiels et parallèles).
- Architecture des ordinateurs.
- Langages et paradigmes de programmation.
- Génie logiciel.
- Autres : mathématiques discrètes, programmation, systèmes d'exploitation, compilation, etc.

Cours enseignés et résultats des évaluations¹

Nombre total de cours enseignés	71
Nombre de cours de premier cycle	48
Nombre de cours de cycles supérieurs	23
Nombre de sigles de cours différents	25
Nombre de cours différents de premier cycle	17
Nombre de cours différents de cycles sup.	8
Nombre de nouveaux cours (enseignés pour la 1 ^{ère} fois à l'UQAM)	7
Nombre de cours évalués	56
Nombre de cours évalués avec moyenne ≥ 4 sur 5 (ou l'équivalent)	53
Nombre de mentions d'excellence	22
Nombre de sigles différents avec mention d'excellence	8

Matériel pédagogique²

Cours	Notes de cours ou Diapos	Exercices	Site Web
Réalisation et maintenance de logiciels (2015–2017)	1279 diapos	7 séries	MGL7460
Langages de script et langages dynamiques (2016–2018)	1437 diapos	7 séries	INF600A
Modélisation et spécification formelle de logiciels (2008–2009)	307 pages	9 séries	INF3140
Programmation parallèle (2005–2014)	404 pages	8 séries	INF5171 INF7235
Construction et maintenance de logiciels (2005)	74 pages		INF3135
Conception et analyse d'algorithmes (2002–2005)	440 pages	11 séries	INF4100 INF7440
Méthodes formelles (1998–2000)	702 pages	18 séries	MAT3143 MGL7160

¹En informatique, les unités de programme attribuent des **mentions** (lettres) d'excellence aux enseignant.e.s ayant obtenu les meilleures évaluations : 2 (ou 3) par session au 1^{er} cycle, 1 (ou 2) aux cycles supérieurs.

²Depuis 1998, toujours avec sites Web.

Logiciels pédagogiques

- **OclCollections, une bibliothèque des types immuables OCL utilisée dans le cours INF3140 pour illustrer l'utilisation des assertions et des tests unitaires**
- **TypesSpec, une bibliothèque des types immuables Spec utilisée dans le cours INF3140 pour illustrer l'utilisation des assertions et des tests unitaires**
- **Oto, un outil d'aide à la correction des travaux de programmation**
Publications : [GT04, TGP05, GT06, TTS06, TL06, TLS07, TGPS08]
- **biblio, un corpus de maintenance pour le cours INF3135**
Publications : [TZMM06, TMSZ07]
- **MiniCUnit, un cadre de tests pour le langage C**
Publication : [TMSZ07]
- **MPDUnit, un cadre de tests pour le langage MPD**
- **Scripts CGI (formulaires Web) pour utilisation par les étudiant.e.s**
 - Pour obtenir leurs notes par courriel (1996)
 - Pour effectuer des fouilles dans mes bases de données bibliographiques
 - Pour effectuer en ligne le choix d'un sujet (ou d'une date) de présentation orale
- **Outil de vérification de types pour le langage de spécification formelle Spec utilisé dans le cours MAT3143/INF3140**
- **Système interactif d'aide aux exercices pour le cours MAT3143/INF3140**
Publication : [Tre99]

Subventions pour projets d'innovation pédagogique

Remarque : Les subventions qui suivent, qui n'incluent pas les subventions CRSNG, sont en lien avec les «Logiciels pédagogiques» de la section précédente

- Subvention (avec L. Laforest et N. Séguin) du FDP (Fonds de Développement Pédagogique) (2005–06) : “Développement d'une banque de travaux de laboratoires et de scripts de correction pour le cours INF1120.” [5000 \$]
- Subvention (avec V. Makarenkov et Bruno Malenfant) du FDP (Fonds de Développement Pédagogique) (2005–06) : “Développement d'un *corpus de maintenance* pour le cours INF3135.” [5000 \$]
- Subvention (avec A. Pons) du CLI (Comité de Liaison Institutionnel) (2003–04) : “Outil générique et évolutif d'aide à la correction automatique de programmes.” [4550 \$]
- Subvention (avec A. Pons) du FDP (Fonds de Développement Pédagogique) (2003–04) : “Un outil générique et évolutif d'aide à la correction de programmes.” [5000 \$]
- Subvention du FITP (Fonds d'Innovation Techno-Pédagogique) (2000–01) : “Outil d'aide à la correction de spécifications formelles.” [3000 \$]
- Subvention du FDP (Fonds de Développement Pédagogique) (1998–99) : “Un outil de vérification de types pour le langage de spécification formelle utilisé dans le cours MAT3143.” [2683 \$]
- Subvention du FDP (Fonds de Développement Pédagogique) (1997–98) : “Système interactif d'aide aux exercices pour le cours de "Méthodes formelles".” [2909 \$]

Supervision d'étudiant.e.s

	Complétés	En cours
Projet ou stage de 1er cycle	26	
Stage de 2 ^e cycle	4	
DESS et Maîtrise en génie logiciel	10	
Maîtrise en informatique	24	1
Doctorat en informatique	2	

- S. Benabdallah (rapport de projet, Maîtrise en génie logiciel, codirection avec Y. Bendavid), “Développement d’un prototype d’abribus connecté par LiFi”, février 2021.
- I. Ciobanu (mémoire, maîtrise en informatique), “*PpFf* : Une bibliothèque C++ pour le traitement parallèle de flux de données”, janvier 2021.
- E. Haas (mémoire, maîtrise en informatique), “*Game Genesis* Un profil UML pour la rédaction de *Game Design Documents*”, décembre 2019.
- M.S. Sall (rapport de projet technique, DESS en génie logiciel), “Annuaire médical: Un répertoire centralisé pour rechercher des ressources en santé”, mars 2019.
- A. Terrasa (doctorat en informatique, codirection avec J. Privat), “Auto-documentation assistée de logiciels: génération et maintenance de fichiers README avec l’outil *nitreadme*”, décembre 2018.
- P. Girard (rapport de projet, Maîtrise en génie logiciel), “Développement d’un module d’extension Moodle d’aide à la correction de questions de type "Texte long"”, juin 2018.
- J.-D. Frondo (rapport de projet technique, DESS en génie logiciel), “Application de simulation de *fills* pour la Banque Nationale du Canada: *Fill Simulator*”, juin 2018.
- R. Chanoir (mémoire, maîtrise en informatique, co-direction avec J. Privat), “CelluloNit: Une implémentation du modèle acteur en Nit”, décembre 2017.
- C. Misale (doctoral thesis, co-supervised with M. Aldinucci): “*PiCo: A Domain-Specific Language for Data Analytics Pipelines*”, Doctoral School on Science and High Technology, Computer Science Department, University of Torino, May 2017.
- G. Youssef (rapport de projet, maîtrise en génie logiciel) : “Centrale d’alerte Web”, avril 2017.
- F. Thériault (mémoire, maîtrise en informatique), “Conception d’un algorithme pour l’assignation de tâches d’enseignement”, avril 2016.
- A. Demange (mémoire, maîtrise en informatique, co-direction avec N. Moha), “Détection de patrons de conception dans les architectures orientées services”, février 2014.
- J. Tsheke (mémoire, maîtrise en informatique), “Détection de défauts de programmes Java”, janvier 2014.
- S. Poirier (mémoire, maîtrise en informatique, co-direction avec J. Privat): “Extensions parallèles pour le langage Nit”, juin 2013.
- E. Lévesque (mémoire, maîtrise en informatique, co-direction avec I. Khriiss, UQAR): “Adaptation des processus collaboratifs par coordination des changements et migration des instances”, août 2011.
- H. Bagane (mémoire, maîtrise en informatique, co-direction avec N. Séguin): “Analyse des principes du génie logiciel au niveau du développement agile”, mars 2011.
- W. Jendoubi (mémoire, maîtrise en informatique): “Surveillance dynamique de compositions de services Web à l’aide de protocoles de comportement”, octobre 2010.
- I. Chouchani (mémoire, maîtrise en informatique): “Utilisation d’un algorithme génétique pour la composition de services *Web*”, mai 2010.
- **P. Lessard (mémoire, maîtrise en informatique): “Un langage spécifique au domaine pour l’outil de correction de travaux de programmation Oto”, février 2010.**

- A. Bustros (mémoire, maîtrise en informatique): “Un système d’intelligence artificielle pour le jeu de plateau *Axis & Allies*”, juillet 2009.
- A. Chami (mémoire, maîtrise en informatique, co-direction avec A. Salah) : “Vérification de processus BPEL à l’aide de Promela/Spin”, mars 2008.
- R. Ben Younes (mémoire, maîtrise en informatique, co-direction avec G. Bégin): “Étude et mise en oeuvre d’une approche formelle de *model-checking* pour la détection d’intrusions dans les réseaux sans-fil 802.11”, novembre 2007.
- **M. Takim (rapport de projet, maîtrise en génie logiciel, co-direction avec A. Salah) : “Applications *Web* pour l’utilisation des services de l’outil Oto”, février 2007.**
- N. Xia (mémoire, maîtrise en informatique) : “A heuristic algorithm for courses assignment”, août 2006.
- I. Constantineau (mémoire, maîtrise en informatique) : “Vérification de modèle floue”, juillet 2006.
- M. Lalonde (mémoire, maîtrise en informatique, co-direction avec M. Jébrak, Dép. des sciences de la terre et de l’atmosphère) : “Modélisation par automates cellulaires de brèches hydro-thermales”, mars 2006.
- **F. Guérin (mémoire, maîtrise en informatique) : “Oto, un outil générique et extensible pour corriger les travaux de programmation”, octobre 2005.**
- P. Hénaff (rapport de projet, maîtrise en génie logiciel) : “Étude des facteurs influençant le choix d’un processus de développement”, avril 2004.
- **F. Champagne (mémoire, maîtrise en informatique) : “Application de la distance d’édition à la correction de dictées musicales”, août 2003.**
- B. Desjardins (rapport de projet, maîtrise en génie logiciel, co-direction avec L. Martin) : “Réingénierie et modélisation d’un composant Adresse à l’aide d’UML”, décembre 2002.
- **É. Labonté (mémoire, maîtrise en informatique) : “OCETJ : Un outil de correction semi-automatique de programmes Java”, décembre 2002.**
- H. Msheik (mémoire, maîtrise en informatique) : “Code generation for a functional language targeted to a parallel multithreaded machine”, août 2000.
- H. Salman (rapport de projet, maîtrise en génie logiciel, ÉTS) : “Modélisation d’un générateur de code pour un langage fonctionnel ciblé vers une machine parallèle multi-contextes”, août 2000.
- B. Malenfant (mémoire, maîtrise en informatique) : “Sémantique et mise en oeuvre d’un langage fonctionnel via traduction en pi-calcul”, décembre 1998.
- J.-P. Tovessi (mémoire, maîtrise en informatique) : “Solution d’une équation différentielle elliptique sur une machine parallèle à l’aide d’échanges de messages”, sept. 1998.
- S. Hadj Messaoud (rapport de projet, maîtrise en génie logiciel) : “Spécification de service de télécommunications : Génération de tests de services et interactions conflictuelles”, fév. 1998.

SERVICES À LA COLLECTIVITÉ

- Membre du comité d'experts externes pour l'évaluation périodique des programmes de 1^{er} cycle en informatique de l'Université de Sherbrooke : 2021
- **Directeur de la maîtrise en informatique de gestion (MIG) : 2019–2020**
Modification majeure de la MIG pour la transformer en «Maîtrise en informatique pour l'intelligence et la gestion des données»
- Directeur de l'unité de programme des cycles supérieurs en génie logiciel : 2018
- Organisation et animation d'une journée d'études sur l'IA : 2018
Activité de «remue-méninges» départemental suite à laquelle deux projets ont été amorcés : ajout d'une concentration en IA à la Maîtrise en informatique et modification majeure de la MIG (voir plus haut)
- Membre du comité de programme de la maîtrise en génie logiciel : 2015–2018
- Membre du comité facultaire de promotion (faculté des sciences) : 2015–2017
- Directeur du département d'informatique, membre du comité exécutif, du comité d'évaluation des professeurs, du comité d'engagement, du comité de la recherche : 2009–2014
- Membre du comité de régie de la Faculté des sciences, du comité facultaire de gestion des espaces, du conseil académique : 2009–2014
- Membre du comité d'auto-évaluation du baccalauréat en informatique et génie logiciel : 2011–2014
- Membre (observateur) au comité de génie de la Faculté des sciences : 2009–2014
- Membre du comité de programme du doctorat en informatique : 2008–2009
- Membre du comité de programme du bac. en informatique et génie logiciel : 2007–2009
- Membre du comité d'experts externes pour l'évaluation périodique des programmes de maîtrise et doctorat en informatique de l'Université Laval : 2006
- **Directeur du baccalauréat en informatique et génie logiciel : 2001–2006**
Plusieurs modifications mineures et une **modification majeure**, laquelle a introduit le langage Java dans les cours de base et renforcé la composante génie logiciel, permettant au programme d'obtenir par la suite son accréditation en génie logiciel (2004–09)
- Directeur adjoint à l'enseignement du Département d'informatique : 2003–2005
- **Membre du comité de développement du programme de doctorat en informatique : 2003–2006**
Analyse des programmes existants en Amérique du Nord, définition de la structure du programme et du cadre pédagogique général, révision des (nombreuses!) versions des documents composant le dossier de création de programme, etc.
- **Membre du Comité de soutien pédagogique du Département d'informatique : 2002–2012**
Visites dans les classes, avec la permission de l'enseignant.e, suivies de rapports écrits — *privés et confidentiels*, contenant diverses observations et suggestions — et de rencontres avec les enseignant.e.s
- **Membre du Conseil d'accréditation des programmes d'informatique (CSAC, Computer Science Accreditation Council) de l'ACI (Association Canadienne d'Informatique) : 2000–2006**
Visites d'accréditation, mise à jour des versions *françaises* des critères et procédures d'accréditation, et **adaptation** de ces critères à la réalité Québécoise.
- **Vice-président du Conseil d'accréditation des programmes d'informatique (CSAC) de l'ACI : 2005–2006**
Modification majeure des critères d'accréditation pour prendre en compte le **génie logiciel**, et ce en m'appuyant sur ma connaissance du *Guide to the SWEBOK*
- Membre du *Committee on Software Engineering Issues* de l'ACI (*Association Canadienne d'Informatique*) : 2004–2006
- Membre du Comité conseil sur le génie logiciel de la FIQ (*Fédération Informatique du Québec*) : 2005–2006

- **Participation au projet UQAM-PADES/EPT pour la réforme des programmes universitaires de Guinée 2005–2006**
Évaluation des programmes de licence en mathématiques–informatique et de maîtrise en informatique, visite pour la formation des enseignants au développement de curriculum, proposition de restructuration majeure du maîtrise en informatique, reflétant les corpus de connaissances en informatique et en génie logiciel
- **Participation à la modification de la maîtrise en informatique : 2004–05**
Révision et élaboration de (5) nouveaux descripteurs de cours
- Membre du comité d’experts externes pour l’évaluation périodique du programme de baccalauréat en informatique de l’UQTR : 2002
- **Coordonnateur/organisateur du Concours de programmation UQAM : Hiver 2002 et 2003**
Planification et organisation générale du concours, rencontres avec des professeurs pour préparer des problèmes et leurs solutions, recherche de commandites, etc.
- Membre du Conseil Académique de la Faculté des Sciences : 2001–2002
- Membre du Comité des Études de la Faculté des Sciences : 2001–2002
- **Membre du Comité de mise sur pied du DESS en Bio-informatique : 2001**
- Membre du Comité des Programmes de la Faculté des Sciences : 2000–2001
- Membre du Comité d’évaluation des professeurs du Dépt d’informatique : 1998–99, 2002–03, 2005–06 (substitut)
- **Responsable de l’accréditation du programme d’informatique de gestion auprès du Conseil d’accréditation (CSAC) de l’ACI : 1998–2000**
Préparation du questionnaire détaillé (institution, département, budgets et ressources humaines, corps enseignant, clientèles étudiantes, programmes, cours, etc.), collecte des nombreux documents afférents (règles administratives, plans de cours, matériel pédagogique, examens, etc.), organisation de la visite du comité (deux jours), etc.
- **Adjoint au directeur du module d’informatique de gestion : 1997–1999, 2000–2001**
- Membre du conseil de module du bac. en informatique et génie logiciel (nouveau nom) : 2001–2003
- Membre du conseil de module du bac. en informatique de gestion : 1986–1988, 1997–99, 2000–2001
- Membre du comité d’usagers de l’informatique (CUI) : 1997–1999
- Responsable de la bibliothèque, section informatique : 1997–1998
- Membre du comité de programme de la maîtrise en informatique : 1996–1999
- Membre du conseil de module des certificats en informatique : 1990–1992, 1994–1998

Intérêts de recherche

- Programmation parallèle (langages, applications scientifiques).
- Méthodes formelles de spécification et de vérification de logiciels.
- **Outils d'aide à l'enseignement et à la correction.**

Subventions

- Subvention (avec J. Privat) pour projet patrimonial, LATECE, UQAM (2013–2015) : “Vers l’auto-documentation «parfaite» de systèmes logiciels”. [8 000 \$/année]
- Subvention individuelle de recherche du CRSNG (2012–2017) : “Protocoles de comportement pour composants et services.” [14 000 \$/année]
- Subvention individuelle de recherche du CRSNG (2006–2011) : “Spécification et vérification de services *Web*.” [19 100 \$/année]
- Subvention individuelle de recherche du CRSNG (2001–05) : “Vérification parallèle de modèles et pi-calcul.” [19 000 \$/année]
- Subvention individuelle de recherche du CRSNG (1996–2001) : “Programmation fonctionnelle pour architectures multi-contextes.” [1996–98 : 16 000 \$/année, 1998–99 : 17 600 \$, 2000–01 : 18 480 \$]
- Subvention PAFACC/Fonds institutionnel de recherche — Nouveau chercheur (1994–96) : “Exploitation des machines parallèles multicontextes via l’utilisation de langages fonctionnels.” [1994–95 : 13 928 \$ et 1995–96 : 10 248 \$]

Publications

Note : Dans ce qui suit, les publications indiquées **en gras** portent sur l’enseignement, les outils d’enseignement ou la définition du génie logiciel. Les autres sont liées à mes activités «régulières» de recherche — méthodes formelles, composants et services Web, programmation parallèle, etc.

Journaux

- [MDT⁺18] C. Misale, M. Drocco, G. Tremblay, A.R. Martinelli, and M. Aldinucci. PiCo: High-performance data analytics pipelines in modern C++. *Future Generation Computer Systems*, 87:392–403, May 2018.
- [ADMT18] M. Aldinucci, M. Drocco, C. Misale, and G. Tremblay. Languages for big data analysis. In S. Sakr and A. Zomaya, editors, *Encyclopedia of Big Data Technologies*, pages 1–12. Springer International Publishing AG, Cham, 2018.
- [MDAT17] C. Misale, M. Drocco, M. Aldinucci, and G. Tremblay. A Comparison of Big Data Frameworks on a Layered Dataflow Model. *Parallel Processing Letters*. 27(1), 20 pages, 2017.
- [GFMTG17] J. Gonzalez-Huerta, M. Founi, N. Moha, G. Tremblay, and Y.G. Guéhéneuc. Semantic Analysis of RESTful APIs for the Detection of Linguistic Patterns and Antipatterns. *International Journal of Cooperative Information Systems*, 26(2), May 2017.
- [BTS11] **H. Bagane, G. Tremblay, and N. Séguin. Principes de génie logiciel et développement agile : une analyse. *Génie logiciel*, 98:43–51, Septembre 2011.**
- [MTB⁺10] H. Mili, G. Tremblay, G. Bou Jaoude, É. Lefebvre, L. Elabed, and G. El Boussaidi. Business process modeling languages: Sorting through the alphabet soup. *ACM Computing Surveys*, 43(1), Article 4, 56 pages, November 2010.
- [LTJ10] M. Lalonde, G. Tremblay, and M. Jébrak. A cellular automata breccia simulator (CABS) and its application to rounding in hydrothermal breccias. *Computers & Geosciences*, 36(7):827–838, July 2010.
- [DHT⁺08] N. Desnos, M. Huchard, G. Tremblay, C. Urtado, and S. Vauttier. Search-based many-to-one component substitution. *Journal of Software Maintenance and Evolution: Research and Practice*, 20(5):321–344, Sept.–Oct. 2008.
- [TGPS08] **G. Tremblay, F. Guérin, A. Pons, and A. Salah. Oto, a generic and extensible tool for marking programming assignments. *Software—Practice and Experience*, 38(3):307–333, March 2008.**
- [TC07] **G. Tremblay and F. Champagne. Marking musical dictations using the edit distance algorithm. *Software—Practice and Experience*, 37(2):207–230, Feb. 2007.**
- [TMAG03] G. Tremblay, C.J. Morrone, J.N. Amaral, and G.R. Gao. Implementation of the EARTH programming model on SMP clusters: a multi-threaded language and runtime system. *Concurrency and Computation: Practice and Experience*, 15(9):821–844, Aug. 2003.
- [Tre02] G. Tremblay. Évolution d’un langage pour la programmation parallèle multi-contextes : Threaded-C. *Technique et Science Informatiques*, 21(5):605–634, 2002.
- [WTA01a] C. Wallace, G. Tremblay, and J.N. Amaral. An abstract state machine specification and verification of the location consistency memory model and cache protocol. *Journal of Universal Computer Science*, 7(11):1089–1113, 2001. Special Issue on Abstract State Machines 2001: New Developments and Applications.
- [TM01] G. Tremblay and B. Malenfant. Lenient evaluation and parallelism. *Computer Languages*, 26(1):43–66, June 2001.

- [Tre01c] G. Tremblay. Lenient evaluation is neither strict nor lazy. *Computer Languages*, 26(1):27–41, June 2001.
- [Tre00c] G. Tremblay. Formal methods: Mathematics, computer science, or software engineering? *IEEE Transactions on Education*, 43(4):377–382, Nov. 2000.

Livres et chapitres de livre

- [TP05] G. Tremblay and A. Pons. Software design: An overview. In R.H. Thayer, editor, *Software Engineering, Volume 1: The Development Process (Third edition)*, pages 195–208. IEEE Computer Society Press, Los Alamitos, CA, 2005.
- [Tre04a] G. Tremblay. *Modélisation et spécification formelle des logiciels (édition revue et augmentée)*. Loze-Dion Éditeurs, Inc., Montréal, QC, quatrième trimestre 2004. 635 p.
- [Tre04] G. Tremblay (Associate Editor). Software design. In A. Abran, J.W. Moore, P. Bourque, and R. Dupuis, editors, *Guide to the Software Engineering Body of Knowledge (2004 Version)*, chapter 3, pages 3.1–3.12. IEEE Computer Society Press, Los Alamitos, CA, 2004. <http://www.swebok.org>.
- [Tre01g] G. Tremblay. Software design. In A. Abran, J.W. Moore, P. Bourque, and R. Dupuis, editors, *Guide to the Software Engineering Body of Knowledge (Trial Version)*, chapter 3, pages 35–51. IEEE Computer Society Press, Los Alamitos, CA, 2001.
- [Tre00h] G. Tremblay. *Modélisation et spécification formelle des logiciels*. Loze-Dion éditeur, Inc., Montréal, QC, 2ième trimestre 2000. 611 p.

Comptes rendus de conférences (avec comité de lecture)

- [MDAT17] C. Misale, M. Drocco, G. Tremblay, and M. Aldinucci. *PiCo: A Novel Approach to Stream Data Analytics*. In *Euro-Par 2017 Auto-DaSP Workshop*, Santiago de Compostela, Spain, August 2017.
- [MDAT16] C. Misale, M. Drocco, M. Aldinucci, and G. Tremblay. *A comparison of big data frameworks on a layered dataflow model*. In *9th Int’l Symposium on High-Level Parallel Programming and Applications (HLPP)*, Muenster, Germany, July 2016.
- [PGMGT15] F. Palma, J. Gonzalez-Huerta, N. Moha, Y.-G. Guéhéneuc, and G. Tremblay. Are RESTful APIs Well-designed? Detection of Linguistic (Anti)patterns In *13th International Conference on Service Oriented Computing (ICSOC 2015)*, 2015.
- [PMTG14] F. Palma, N. Moha, G. Tremblay, and Y.-G. Guéhéneuc. Specification and detection of SOA antipatterns in web services. In *8th European Conference on Software Architecture (ECSA 2014)*, pages 58–73, Vienna, Austria, Aug. 2014. Springer-Verlag, LNCS-8627.
- [DMT13] A. Demange, N. Moha, and G. Tremblay. Detection of SOA patterns. In *Int. Conf. on Service-Oriented Computing (ICSOC)*, pages 114–130. Springer-Verlag, LNCS-8274, Berlin, Germany, Dec. 2013.
- [BT13] R. Ben Younes and G. Tremblay. A possibilistic approach to intrusion detection under imperfect logging protocol. In *Proc. of the 6th Intl. Conf. on Security of Information and Networks (SIN ’13)*, pages 266–271, Aksaray, Turkey, Nov. 2013. ACM Press.
- [STB12] N. Séguin, G. Tremblay, and H. Bagane. Agile Principles as Software Engineering Principles: An Analysis. In C. Wohlin, editor, *XP 2012*, volume 111 of *LNBIP*, pages 1–15, Berlin Heidelberg, 2012. Springer-Verlag.

- [TL11] G. Tremblay and P. Lessard. **A marking language for the Oto assignment marking tool.** In *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education*, pages 148–152, Darmstadt, Germany, June 2011. ACM.
- [STC09] A. Salah, G. Tremblay, and A. Chami. Checking conformity of WS-BPEL processes with respect to interface protocols. In A. Obaid, editor, *Proceedings of the 9th Annual International Conference on New Technologies of Distributed Systems (NOTERE'09)*, pages 289–299, Montréal, Canada, June 2009.
- [BTB08] R. Ben Younes, G. Tremblay, and G. Bégin. Extending Orchids for intrusion detection in 802.11 wireless networks. In D. Benslimane and A. Ouksel, editors, *Proceedings of the 8th International Conference on New Technologies in Distributed Systems (NOTERE 2008)*, volume 1, pages 40–51, Lyon, France, June 2008. ACM Edition.
- [KLTJ08] I. Khriess, E. Lévesque, G. Tremblay, and A. Jacques. Towards adaptability support in collaborative business processes. In P. Kropf, M. Benyoucef, and H. Mili, editors, *2008 International MCETECH Conference on e-Technologies*, pages 34–45, Montréal, Canada, Jan. 2008. IEEE Computer Society.
- [STC08] A. Salah, G. Tremblay, and A. Chami. Behavioral interface conformance checking for WS-BPEL processes. In P. Kropf, M. Benyoucef, and H. Mili, editors, *2008 International MCETECH Conference on e-Technologies*, pages 253–256, Montréal, Canada, Jan. 2008. IEEE Computer Society.
- [DHU⁺07] N. Desnos, M. Huchard, C. Urtado, S. Vauttier, and G. Tremblay. Automated and unanticipated flexible component substitution. In *The 10th International ACM SIGSOFT Symposium on Component-Based Software Engineering (CBSE '07)*, pages 33–48, Springer-Verlag, LNCS-4608, Medford, MA, July 2007.
- [TMSZ07] G. Tremblay, B. Malenfant, A. Salah, and P. Zentilli. **Introducing students to professional software construction: A “Software construction and maintenance” course and its maintenance corpus.** In *Proceedings of the 12th Annual Conference on Innovation and Technology in Computer Science Education*, pages 176–180. ACM, June 2007.
- [MFB⁺06] H. Mili, M. Frendi, G. Bou Jaoude, G. Tremblay, and L. Martin. Classifying business processes for domain engineering. In *18th IEEE International Conference on Tools with Artificial Intelligence*, pages 397–406, Washington, DC, Nov. 2006. IEEE Computer Society.
- [MGTD06] H. Mili, R. Godin, G. Tremblay, and W. Dorfeuille. Towards a methodology for designing compensation processes in long-running business transactions. In L. Logrippo, H. Mili, and A. Salah, editors, *Proceedings of the 2006 Montreal Conference on eTechnologies*, pages 137–146, Montréal, Canada, May 2006.
- [TGP05] G. Tremblay, F. Guérin, and A. Pons. **A generic and extensible tool for marking programming assignments.** In T.C. Mongomerie and J.R. Parker, editors, *The IASTED International Conference on Education and Technology (ICET 2005)*, pages 55–60, Calgary, AB, Canada, July 2005. ACTA Press.
- [TC05] G. Tremblay and J. Chae. Towards specifying contracts and protocols for Web services. In H. Mili and F. Khendek, editors, *MCeTech Montreal Conference on eTechnologies*, pages 73–85, Montréal, Canada, Jan. 2005.
- [CTB05] A.-É. Caillot, G. Tremblay, B. Ben Tamrout, A. Obaid, and H. Mcheick. Web service composition as a function cover problem. In H. Mili and F. Khendek, editors, *MCeTech Montreal Conference on eTechnologies*, pages 61–71, Montréal, Canada, Jan. 2005.
- [MBBT04] H. Mili, F. Bédard, G. Bou Jaoude, and G. Tremblay. Business process modeling for light-weight inter-enterprise system integration. In A. Frew, editor, *Information and Communication Technologies in Tourism (ENTER 2004)*, pages 380–390, Cairo, Egypt, January 2004. International Federation for IT and Travel & Tourism (IFITT), Springer.

- [TL03b] G. Tremblay and É. Labonté. Semi-automatic marking of Java programs using JUnit. In *International Conference on Education and Information Systems: Technologies and Applications (EISTA '03)*, pages 42–47, Orlando, FL, Jul. 2003. International Institute of Informatics and Systemics.
- [TC02] G. Tremblay and F. Champagne. Automatic marking of musical dictations by applying the edit distance algorithm on a symbolic music representation. In G. Haus & M. Longari, editor, *Proceedings First International Conference MAX2002 Musical Application Using XML*, pages 11–17, Milan, Italy, Sept. 2002. IEEE Computer Society Technical Committee on Computer Generated Music.
- [WTA02] C. Wallace, G. Tremblay, and J.N. Amaral. On the tamability of the location consistency memory model. In *International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA '02)*, pages 1542–1547, Las Vegas, NE, June 2002.
- [MATG01] C.J. Morrone, J.N. Amaral, G. Tremblay, and G.R. Gao. A multi-threaded runtime system for a multi-processor/multi-node cluster. In R.D. Kent and T.W. Sands, editors, *High Performance Computing Systems and Applications (HPCS '2001)*, pages 229–242, June 2001. Kluwer Academic Publishers, Norwell, MA, 2003.
- [Tre01a] G. Tremblay. Évolution d'un langage pour la programmation parallèle multi-contextes : Threaded-C. In C. Cérin and G. Utard, editors, *RENPAR '13*, pages 133–138, Paris, France, avril 2001. ASTI '2001.
- [WTA01] C. Wallace, G. Tremblay, and J.N. Amaral. Specification and verification of the LC cache consistency protocol (extended abstract). In R. Moreno-Diaz and A. Quesada-Arencibia, editors, *Formal Methods and Tools for Computer Science, EUROCAST 2001*, pages 275–278. Universidad de Las Palmas de Gran Canaria, Feb. 2001.
- [Tre00] G. Tremblay. Formal methods: Mathematics, computer science, or software engineering? In *Proceedings of the International Conference on Software Engineering Education and Training (CSEET '00)*, pages 273–282. IEEE Computer Society, 2000.
- [Tre99] G. Tremblay. Exercises for teaching logic in a formal methods course: Formalizing ERDs. In *ACM SIGCSE Technical Symposium*, pages 100–104. ACM, 1999.
- [Tre98] G. Tremblay. An undergraduate course in formal methods: “Description is our business”. In *ACM SIGCSE Technical Symposium*, pages 166–170. ACM, 1998.
- [CZT97] O. Cherkaoui, S. Znaty, and G. Tremblay. Reusability and interoperability in SNMP-based network management: From concepts to applications. In *International Conference on Telecommunications*, Melbourne, Australia, April 1997.
- [Tre97] G. Tremblay. Langages fonctionnels non-stricts : De la paresse et de l'indulgence. In *JFLA 97 – Journées Francophones des Langages Applicatifs*, pages 189–208. INRIA #16 (Collection Didactique), 1997.
- [Tre96a] G. Tremblay. Compilation de langages paresseux par analyse de stricticité arrière : Utilité et limites. In *JFLA 96 – Journées Francophones des Langages Applicatifs*, pages 225–244. INRIA #15, 1996.
- [TG95] G. Tremblay and G.R. Gao. The impact of laziness on parallelism and the limits of strictness analysis. In A.P. Wim Bohm and J. T. Feo, editors, *High Performance Functional Computing*, pages 119–133. Lawrence Livermore National Laboratory, CONF-9504126, 1995.

Posters (avec publication dans compte rendu)

- [JTS11] W. Jendoubi, G. Tremblay, and A. Salah. Interface expressions monitoring for BPEL processes (*poster*). In *7th International Conference on Web Information Systems and Technologies (WEBIST 2011)*, pages 131–134, Noordwijkerhout, Netherlands, May 2011.
- [TLS07] G. Tremblay, L. Laforest, and A. Salah. Extending a marking tool with simple support for testing (*Poster*). In *Proceedings of the 12th Annual Conference on Innovation and Technology in Computer Science Education*, page 313. ACM, June 2007.

Rapports techniques

- [MDAT16] C. Misale, M. Drocco, M. Aldinucci, and G. Tremblay. *A survey of big data frameworks on a layered dataflow model*. Rapport de recherche Latece 2016-1, Dép. d’informatique, UQAM, Mai 2016.
- [TPT15b] A. Terrasa, J. Privat, and G. Tremblay. Documentation generators support for program comprehension: Where are we? Rapport de recherche Latece 2015-2, Dép. d’informatique, UQAM, Déc. 2015.
- [TA15] G. Tremblay and M. Aldinucci. An internal DSL for building FastFlow skeletons using a C++ fluent interface. Rapport de recherche Latece 2015-1, Dép. d’informatique, UQAM, Juin 2015.
- [GTL10] F. Guérin, G. Tremblay, and P. Lessard. *Oto, un outil d’aide à la correction de programmes : Guide d’utilisation à l’intention des enseignants et des auxiliaires d’enseignement — Oto 2*. Note interne de recherche, Dép. d’informatique, UQAM, <http://oto.uqam.ca/documents/manuel-enseignant-2.pdf>, Décembre 2010.
- [TFDH07] G. Tremblay, L. Fabresse, C. Dony, and M. Huchard. ébauche d’une sémantique opérationnelle pour le langage SCL. Note interne de recherche, mars-juin 2007.
- [TDU⁺07] G. Tremblay, N. Desnos, C. Urtado, S. Vauttier, and M. Huchard. Ports primitifs et composés : Une description formelle (exécutable) des propriétés de cohérence et complétude. Note interne de recherche, Mai 2007.
- [TL06] G. Tremblay and L. Laforest. *Oto, un outil d’aide à la correction de programmes : Guide pratique pour la conception et la correction de travaux de laboratoires*. Note interne de recherche, Dép. d’informatique, UQAM, Novembre 2006. <http://oto.uqam.ca/documents/guide-labos.pdf>
- [TTS06] G. Tremblay, M. Takim, and A. Salah. *Oto, un outil d’aide à la correction de programmes : Guide d’utilisation destiné aux étudiants*. Note interne de recherche, Dép. d’informatique, UQAM, Septembre 2006. <http://oto.uqam.ca/documents/manuel-oto-etudiant.pdf>
- [GT06] F. Guérin and G. Tremblay. *Oto, un outil d’aide à la correction de programmes : Guide d’utilisation à l’intention des enseignants et des auxiliaires d’enseignement*. Note interne de recherche, Dép. d’informatique, UQAM, Mai 2006.
- [TZMM06] G. Tremblay, P. Zentilli, B. Malenfant, and V. Makarenkov. *BIBLIO un logiciel pour la gestion de prêts de livre*. Note interne de recherche, Dép. d’informatique, UQAM, Mars 2006. <http://www.labunix.uqam.ca/~tremblay/INF3135/Biblio/systeme-gestion-livres.pdf>
- [TS05] G. Tremblay and A. Salah. Spécification et vérification formelle de services *Web* : de l’orchestration à la chorégraphie. Note interne de recherche, Dép. d’informatique, UQAM, Septembre–octobre 2005.
- [MTBC04] H. Mili, G. Tremblay, R. Bentamrout, and A.-É. Caillot. Adding semantics to Web service descriptions. Rapport technique LATECE, Dépt d’Informatique, UQAM, Octobre 2004.

- [GT04] F. Guérin and G. Tremblay. **Oto : Un outil générique et évolutif d'aide à la correction de programmes : Langage de scriptage.** Note interne de recherche, Dépt d'informatique, UQAM, Juin 2004.
- [TG04] G. Tremblay and F. Guérin. **Oto : Un outil générique et évolutif d'aide à la correction de programmes : Cas d'utilisation.** Note interne de recherche, Dépt d'informatique, UQAM, Juin 2004.
- [Tre04a] G. Tremblay. **Du bac en informatique de gestion au bac en informatique et génie logiciel.** Répartie, Bulletin du département d'informatique de l'UQAM, vol. 10, no. 2, Été 2004. pages 2–4.
- [MBL+03] H. Mili, G. Bou Jaoude, É. Lefebvre, G. Tremblay, and A. Petrenko. **Business process modeling languages: Sorting through the alphabet soup.** Rapport de recherche, Dépt d'Informatique, UQAM, Jan. 2004.
- [WTA01a] C. Wallace, G. Tremblay, and J.N. Amaral. **The location consistency memory model and cache protocol: Specification and verification.** Technical Report 01-01, Computer Science Department, Michigan Technological University, Houghton, MI, June 2001.
- [Tre01f] G. Tremblay. **A VDM-style specification of the lattice building algorithm.** Note interne de recherche, avril 2001.
- [TTM+00] G. Tremblay, K.B. Theobald, C.J. Morrone, M.D. Butala, J.N. Amaral, and G.R. Gao. **Threaded-C language reference manual (release 2.0).** CAPSL Technical Memo 39, University of Delaware, June 2000.
- [Tre00f] G. Tremblay. **Threaded-C release 2.0: Motivation, description, and rationale.** CAPSL Technical Note 09, University of Delaware, June 2000.
- [Tre00a] G. Tremblay. **Knowledge area description for software design (versions 0.1, 0.50, 0.60, 0.70, 0.80).** Part of the **Guide to the Software Engineering Body of Knowledge (Stone Man Version)**, <http://www.swebok.org>, 1999–00.
- [Tre99d] G. Tremblay. **On choosing a good notation for a formal methods course: Making the concepts explicit.** Rapport Technique 99-07, Dépt d'informatique, UQAM, Déc. 1999.
- [Tre99h] G. Tremblay. **SWEBOK knowledge area jump-start document for coding and testing.** Note interne au projet SWEBOK, mars 1999.
- [Tre99a] G. Tremblay. **À propos des cours de programmation.** Document soumis au comité conjoint bac–certificat pour la restructuration des cours de base, 9 pages, mai 1999.
- [Tre99] Guy Tremblay. **La correction des travaux de programmation et la détection des cas de plagiat : Qu'est-ce qui se fait ailleurs?** Document soumis au comité conjoint bac–certificat pour la restructuration des cours de base, 11 pages, mai 1999.
- [BDA+98] P. Bourque, R. Dupuis, A. Abran, J.W. Moore, L. Tripp, J. Shyne, B. Pflug, M. Maya, and G. Tremblay. **Guide to the software engineering body of knowledge — a straw man version.** Technical report, Dépt d'Informatique, UQAM, Sept. 1998.
- [TM98] G. Tremblay and B. Malenfant. **Sémantique parallèle d'un langage fonctionnel indulgent.** Note interne de recherche, sept. 1998.
- [Tre97a] G. Tremblay. **Compilation de pH pour une machine parallèle multi-contextes.** Note interne de recherche, Dépt d'informatique, UQAM, juil. 1997.
- [Tre96b] G. Tremblay. **Modélisation LOTOS des différentes formes d'évaluation fonctionnelle.** Rapport de recherche, Dépt d'Informatique, UQAM, oct. 1996.

- [Tre91] G. Tremblay. Strictness analysis based on forward abstract interpretation cannot express head-strictness. ACAPS Technical Note 33, School of Computer Science, McGill University, Montréal, QC, Sept. 1991.
- [TG89] G. Tremblay and G.R. Gao. A formal operational model for data-driven program tuples. ACAPS Technical Memo 08, School of Computer Science, McGill University, Montreal, QC, April 1989.
- [GT88] G.R. Gao and G. Tremblay. Efficient demand-driven evaluation on argument-fetching dataflow architecture. ACAPS Technical Memo 05, School of Computer Science, McGill University, Montreal, QC, Oct. 1988.

Communications et séminaires

- [Tre20] G. Tremblay. Programmation fonctionnelle : «Retour vers le futur»? Séminaire Latece, UQAM, 23 juin 2020.
- [Tre19b] G. Tremblay. «Honni soit qui mal y science»: A little stroll through science, bad science... and statistics. Dept. of CS & SE, Concordia University, 29 octobre 2019.
- [Tre19a] G. Tremblay. «Honni soit qui mal y science» : Petite balade dans la science, la malscience... et les statistiques. Séminaire Latece, UQAM, 19 juin 2019.
- [Terrasa *et al.*, 2018] Terrasa, A., Privat, J. et Tremblay, G. (2018). Using natural language processing for documentation assist. Dans *AAAI Technical Report WS-18-14, The Workshops of the Thirty-Second AAAI Conference on Artificial Intelligence*, 787–790. The AAAI Press.
- [Tre17] G. Tremblay. *PRuby, una libreria per la programmazione parallela (anglais/italien)*. Alpha Group, Dipartimento di Informatica, Università degli Studi di Torino. 4 juillet 2017.
- [Tre16a] G. Tremblay. «Towards a Data Analytics API targeted to FastFlow.» Alpha Group, Dipartimento di Informatica, Università degli Studi di Torino. 20 juin 2016.
- [Tre15c] G. Tremblay. *Extending FastFlow with a DSL: Why and How? A look at some alternative approaches*. Alpha Group, Dipartimento di Informatica, Università degli Studi di Torino. 30 mars 2015.
- [Tre15b] G. Tremblay. *Ruby FF: A Ruby implementation of FastFlow*. Alpha Group, Dipartimento di Informatica, Università degli Studi di Torino. 25 février 2015.
- [Tre15a] G. Tremblay. *Implementazione di Linguaggi Specifici di Dominio Interni con Ruby: Un'introduzione (anglais/italien)*. Dipartimento di Informatica, Università degli Studi di Torino. 15 janvier 2015.
- [Tre14b] G. Tremblay. Ruby =~/[TB]DD/. Séminaire Latece, UQAM, 5 novembre 2014.
- [Tre14a] G. Tremblay. Patrons et anti-patrons de présentation. Séminaire Latece, UQAM, 5 février 2014.
- [Tre12b] G. Tremblay. Travail intellectuel et recherche : «Des chemins moins fréquentés» (version révisée/augmentée). Séminaires des Doctorats en informatique et en informatique cognitive, UQAM, 20 février 2013.
- [Tre12] G. Tremblay. Travail intellectuel et recherche : «Des chemins moins fréquentés». Séminaire Latece, UQAM, 20 décembre 2012.
- [Tre10] G. Tremblay et P. Lessard. L'outil de correction Oto 2: Fonctionnalités et mise en oeuvre. Séminaire Latece, UQAM, 15 décembre 2010.**
- [Tre07b] G. Tremblay. Services *Web* : De l'orchestration à la chorégraphie. LIRMM, Montpellier, France, mars 2007.

- [Tre07a] G. Tremblay. Une introduction aux services *Web*. LIRMM, Montpellier, France, février 2007.
- [TG05] G. Tremblay et F. Guérin. **Oto, un outil générique et extensible pour corriger les travaux de programmation.** École Supérieur d'Ingénierie Léonard de Vinci (ESILV), Paris, France, décembre 2005.
- [GT05] F. Guérin et G. Tremblay. **Oto, un outil extensible d'aide à la correction de programmes.** 73ième Congrès de l'ACFAS, Session "Informatique et génie logiciel", Université du Québec à Chicoutimi, Chicoutimi, Mai 2005.
- [Tre03] G. Tremblay. Une introduction à la vérification de modèles. Séminaire du doctorat en informatique cognitive. Dépt d'Informatique, UQAM, Oct. 2003.
- [CT03] F. Champagne et G. Tremblay. **Application de la distance d'édition à la correction de dictées musicales.** Dépt d'Informatique, UQAM, Oct. 2003.
- [LT03] É. Labonté et G. Tremblay. **OCETJ : un outil d'aide à la corrections de programmes basé sur JUnit ... et ses implications possibles pour nos cours de base.** Dépt d'Informatique, UQAM, Mai 2003.
- [Tre03a] G. Tremblay. **Accreditation of Software Engineering Programs by the CSAC.** Canadian Undergraduate Software Engineering Conference, Montréal, Qué, Jan. 2003.
- [Tre02b] G. Tremblay. An introduction to model checking. Dept of Computing Science, University of Alberta, Edmonton, CA, Jul. 2002.
- [LT02] E. Labonté and G. Tremblay. **Système d'aide à la correction de programmes Java.** 70ième Congrès de l'ACFAS, Session "Informatique et génie logiciel", Université Laval, Québec, Mai 2002.
- [CT02] F. Champagne and G. Tremblay. **Application de l'algorithme de distance d'édition à des segments musicaux.** 70ième Congrès de l'ACFAS, Session "Informatique et génie logiciel", Université Laval, Québec, Mai 2002.
- [Tre01b] G. Tremblay. Évolution d'un langage pour la programmation parallèle multi-contextes : Threaded-C. Dépt d'Informatique, UQAM, Sept. 2001.
- [Tre01g] G. Tremblay. Parallel programming models: Where does Threaded-C stand? Dept of ECE, University of Delaware, Aug. 2001.
- [Tre01d] G. Tremblay. An introduction to formal methods. Michigan Technological Universit, Feb. 2001.
- [Tre00i] G. Tremblay. Threaded-C: Where is it coming from? where is it going? Dept of ECE, University of Delaware, Feb. 2000.
- [Tre99f] G. Tremblay. **Méthodes formelles : Mathématiques, informatique, ou génie logiciel?** Institut Francophone d'Informatique, Vietnam, Hanoi, Nov. 1999.
- [Tre98b] G. Tremblay. **Méthodes formelles : Mathématiques, informatique, ou génie logiciel?** Dépt d'Informatique, UQAM, Sept. 1998.
- [MT98] B. Malenfant and G. Tremblay. Sémantique parallèle d'un langage fonctionnel indulgent. 66ième Congrès de l'ACFAS, Session "Programmation parallèle et génie logiciel", Université Laval, Québec, Mai 1998.
- [Tre97c] G. Tremblay. Langages fonctionnels non-stricts : De la paresse et de l'indulgence. INRIA Rocquencourt, Domaine de Voluceau, France, Juil. 1997.
- [Tre96c] G. Tremblay. Machines parallèles à mémoire distribuée : Protocoles de cohérence de caches et modèles de consistance mémoire. Groupe de Télé-informatique, Dépt d'Informatique, UQAM, Oct. 1996.

- [Tre96b] G. Tremblay. Langages fonctionnels non-stricts : De la paresse à l'indulgence. Dépt d'Informatique, UQAM, Mars 1996.
- [Tre95] G. Tremblay. Mise en oeuvre parallèle des langages fonctionnels paresseux : Une approche basée sur la propagation abstraite des demandes,. DIRO, Univ. de Montréal, Nov. 1995.
- [Tre95a] G. Tremblay. Mise en oeuvre parallèle des langages fonctionnels paresseux : Une approche basée sur la propagation abstraite des demandes (I et II). Séminaire de Logique et Informatique, Dépt d'Informatique, UQAM, Fév. 1995.
- [Tre91b] G. Tremblay. Strictness analysis of lists: Why abstract interpretation cannot recognize head-strictness. ACAPS Seminar, School of CS, Univ. McGill, Sept. 1991.
- [Tre90a] G. Tremblay. Efficient implementation of lazy evaluation without abstract interpretation. ACAPS Seminar, School of CS, Univ. McGill, Jan. 1990.
- [Tre89a] G. Tremblay. Abstract interpretation: An informal introduction. ACAPS Seminar, School of CS, Univ. McGill, Déc. 1989.
- [Tre89b] G. Tremblay. Functional programming ... in the twilight zone: Dealing with I/O and resource management. ACAPS Seminar, School of CS, Univ. McGill, Oct. 1989.
- [Tre89f] G. Tremblay. A new functional programming language: Haskell. ACAPS Seminar, School of CS, Univ. McGill, Jun. 1989.
- [Tre89c] G. Tremblay. Higher-order functions and non-strict evaluation: Why they make a difference. ACAPS Seminar, School of CS, Univ. McGill, May 1989.
- [Tre89d] G. Tremblay. Lazy data structures on the argument-fetching dataflow architecture. ACAPS Seminar, School of CS, Univ. McGill, May 1989.
- [Tre89e] G. Tremblay. Mapping functional quads into argument-fetching dataflow graphs — parts I and II. ACAPS Seminar, School of CS, Univ. McGill, Jan. 1989.
- [Tre88] G. Tremblay. A hybrid evaluation model combining data-driven and demand-driven evaluation. ACAPS Seminar, School of CS, Univ. McGill, Oct. 1988.

Autres (colonne invitée dans revue)

- [Tre01d] G. Tremblay. Software design knowledge and Vincenti's categories of engineering knowledge. *IEEE Annals for the History of Computing—Anecdotes Column*, 23(1):76–79, Jan.–Mar. 2001.

Essai de maîtrise et thèse de doctorat

- [TG94] G. Tremblay. *Parallel Implementation of Lazy Functional Languages using Abstract Demand Propagation*. PhD thesis, School of Computer Science, McGill University, Nov. 1994.
- [TG84] G. Tremblay. *Strategies for teaching undergraduate programming*. Master's Essay, Departement of Computer Science, Faculty of Mathematics, University of Waterloo, Dec. 1984.