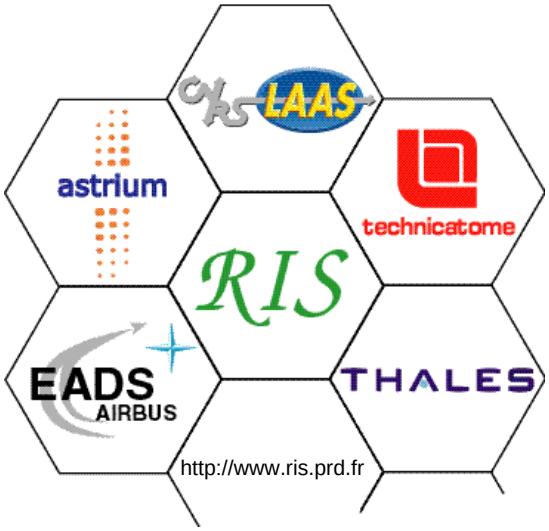

Réseau d'Ingénierie de la Sûreté de fonctionnement

réunion n°1 du GT Logiciel Libre et Sûreté de Fonctionnement
31 mai 2001 – LAAS-CNRS

Programme	
10 h 00-10 h 30	Présentation et objectifs du groupe
10 h 30-12 h 00	Positionnement de chacun par rapport aux Logiciels Libres : 5 minutes/personne
12 h 00-13 h 15	*Déjeuner*
13 h 15-14 h 15	Identification des compétences du groupe, classement préliminaire des thèmes
14 h 15-15 h 15	Mise en place de la logique de travail
15 h 15-15 h 30	*Pause*
15 h 30-16 h 15	Mode de fonctionnement
16 h 15-16 h 30	Préparation de la prochaine réunion et Synthèse

Participants :

Membres du GT :

B. Bérard (LSV), E. Conquet (Astrium), P. Coupoux (Technicatome), Y. Crouzet (LAAS-CNRS), P. David (Astrium), C. Desquilbet (CEAT), Y. Garnier (SNCF), G. Mariano (INRETS), V. Nicomette (LAAS-CNRS), Y. Paindaveine (Commission Européenne), J-M. Tanneau (THALES), H. Waeselynck (LAAS-CNRS).

Invités :

J. Arlat (LAAS-CNRS), T. Vardanega (ESA).

Absents ou excusés :

S. Goiffon (EADS Airbus), I. Puaut (IRISA), J-L. Terraillon (ESA).

1. Présentation et objectifs du groupe

Philippe David présente un bref compte-rendu de l'Atelier Thématique « logiciel libre et sûreté de fonctionnement » qui s'était tenu en décembre 2000, et qui avait montré le fort intérêt des membres fondateurs du *RIS* pour ce thème. Ceci a motivé le lancement du présent Groupe de Travail, d'une durée de vie de 18 mois, qui sera conjointement animé par Philippe David (Animateur), et Hélène Waeselynck (Rapporteur). Les réunions du GT s'effectueront selon une périodicité de 1,5 à 2 mois. L'objectif est de publier une monographie, à la rédaction de laquelle tous les membres du groupe contribueront. Les résultats du GT seront également présentés lors d'une Journée *RIS*.

Jean Arlat, Coordinateur du *RIS*, rappelle le contexte du Réseau d'Ingénierie de la Sûreté de Fonctionnement. Historiquement, cette structure de type " Réseau " résulte de la volonté des membres fondateurs de poursuivre la coopération privilégiée établie dans le cadre du Laboratoire d'Ingénierie de la Sûreté de fonctionnement (*LIS*). Cette nouvelle formule est aussi destinée à favoriser l'extension de la coopération à une participation industrielle et académique élargie au-delà des membres fondateurs.

La formule des GT étend le type de travail effectué dans le cadre de l'action fédératrice *LIS* menée sur les COTS (Commercial-Off-The-Shelf). Cette action avait notamment donné lieu à la publication d'une monographie[1] sur la problématique posée par l'intégration de composants préalablement existants, d'origine commerciale ou non.

2. Positionnement de chacun par rapport aux logiciels libres

Il avait été demandé à chaque participant de préparer une courte intervention pour se situer par rapport à la thématique du GT (ci-dessous par ordre d'intervention).

Hélène Waeselynck, Yves Crouzet, Vincent Nicomette (LAAS-CNRS)

Les travaux du groupe de recherche « Tolérance aux fautes et sûreté de fonctionnement informatique » du LAAS-CNRS sont structurés selon quatre grands thèmes : prévention des fautes, tolérance aux fautes, élimination des fautes, prévision des fautes. La sûreté de fonctionnement des systèmes informatiques est définie comme un concept générique, qui englobe les propriétés de disponibilité, fiabilité, etc., mais aussi de sécurité par rapport aux manipulations non-autorisées de l'information.

Les compétences apportées au sein du GT se situent donc dans les moyens – obtention et validation – de la sûreté de fonctionnement. On peut mentionner quelques travaux en cours spécifiquement liés à la problématique des COTS et des logiciels libres : 1) l'étalonnage (*benchmarking*) de la sûreté de fonctionnement, et 2) la caractérisation des modes de défaillance de composants logiciels. Dans le premier cas, il s'agit d'aider à la sélection de composants (COTS ou logiciels libres), en fournissant des bases objectives pour comparer les alternatives offertes. Dans le deuxième cas, on analyse finement les modes de défaillance d'un composant sélectionné, afin de concevoir des parades architecturales (techniques d'empaquetage, ou *wrapping*). Etalonnage et caractérisation font appel à des méthodes expérimentales basées sur le test et l'injection de fautes. Les composants ainsi évalués sont des micro-noyaux (Chorus, LynxOS), des systèmes d'exploitation à caractère générique (Windows 2000, Linux) et des intergiciels de type CORBA (ORBacus).

Béatrice Bérard (LSV)

Le Laboratoire Spécification et Vérification (LSV) est associé au CNRS depuis 1997. Il regroupe 29 personnes, autour du double thème de la spécification et de la vérification du logiciel. Les compétences du laboratoire en matière de méthodes de vérification formelle couvrent à la fois la vérification de modèles (*model-checking*) et la démonstration automatique (systèmes déductifs). Elles ciblent différentes catégories de logiciels critiques : logiciels de contrôle/commande réactifs, temps-réel, hybrides (continu/discret), protocoles de communication. Des études ont notamment été menées sur des applications du domaine de l'énergie (EDF) ou des télécommunications (Alcatel).

Dans ces exemples, la mise en œuvre a posteriori de méthodes de vérification a nécessité une formalisation du problème, basée sur l'analyse du cahier des charges et du code (ou des algorithmes fonctionnels à partir desquels le code a été produit), avec l'aide d'experts du domaine. L'accès au code source est crucial lors de cette étape de formalisation : dans le cas contraire, on n'a aucune garantie quant à la représentativité des modèles formels sur lesquels porte la vérification (ex : existence de fonctions cachées dans le code).

En conclusion, la disponibilité des sources est une condition nécessaire (bien que non suffisante) pour la vérification formelle, d'où l'intérêt pour les logiciels libres.

[1] « Composants logiciels et sûreté de fonctionnement : intégration de COTS », sous la direction de Jean Arlat (15 auteurs), Hermes Science Publications, juin 2000, ISBN 2-7462-0146-1.

Yseult Garnier (SNCF)

Le département « Signalisation » de la SNCF effectue l'évaluation de logiciels critiques (contrôle de vitesse, postes d'aiguillage) par rapport aux normes européennes CENELEC en vigueur dans le domaine ferroviaire. Dans ce domaine, les logiciels peuvent être classés en trois couches :

- logiciel de base (non critique), pour lequel des COTS sont utilisés (Windows NT) ;
- logiciel applicatif (critique) ;
- logiciel de paramètres (critique).

Pour l'instant, aucun logiciel libre n'est utilisé, même dans le cas des logiciels de base. Le problème serait de faire évoluer les normes qui, si elles mentionnent les COTS, ne prennent pas en compte les logiciels libres. Aujourd'hui, on ne sait donc pas comment homologuer des systèmes incluant des logiciels libres. Il faudrait également définir l'impact de leur utilisation sur le cycle de vie du système. Par exemple, comment adapter une procédure d'enregistrement des versions du LL ? Dans le cas d'un COTS, le composant est maintenu par le fabriquant, ce qui offre une certaine garantie. Dans le cas d'un LL, il faut pouvoir identifier, et valider, ce qui a été modifié.

Georges Mariano (INRETS)

L'INRETS est un institut de recherche appliquée sur les transports, qui effectue également des missions d'expertise dans ce domaine. L'Unité de Recherche « Evaluation des Systèmes de Transport Automatisés et de leur Sécurité » (ESTAS) développe des activités dans le domaine de la sécurité des transports guidés, c'est-à-dire essentiellement le transport ferroviaire.

Les expertises réalisées (homologation) portent en général plus sur l'aspect « processus » que sur l'aspect « produit ». Comme indiqué par Yseult Garnier, il n'y a pas actuellement de logiciels libres dans les applications ferroviaires, mais c'est une question qu'on peut se poser.

Une autre question concerne l'utilisation de logiciels libres dans les environnements de développement d'applications critiques. Suite à l'introduction de la méthode B dans le domaine ferroviaire, des recherches menées à ESTAS ont conduit à développer des outils support, qui manipulent les spécifications formelles. Toute l'infrastructure de développement de ces outils est basée sur des logiciels libres (gcc, Linux, Caml, …), et les outils eux-mêmes sont *open-source*. Ceci est perçu comme apportant une souplesse de travail, et comme un moyen de faciliter des échanges avec d'autres laboratoires, pour coopérer autour de plate-formes expérimentales. Le problème est maintenant celui de l'utilisation de tels outils dans le cadre de projets industriels.

Yves Paindaveine (Commission Européenne)

Y. Paindaveine est *Project Officer* rattaché à l'Unité « Applications liées à la santé » de la Direction « Technologies de la Société de l'Information : Systèmes et services pour le citoyen ».

Le logiciel libre est devenu une des priorités du Programme de Travail de l'IST. Ceci résulte de l'action de plusieurs groupes, dont :

- l'ISTAG (groupe qui conseille la Commission sur le contenu et les orientations du programme IST),
- un groupe de travail sur les logiciels libres[2] créé à l'initiative de la Direction Générale IS (cf. http://eu.conecta.it/),
- l'initiative politique *e*Europe lancée en décembre 1999, qui mentionne les logiciels libres dans deux actions (*secure networks*, et *electronic access to public services*),
- M. Erkki Liikanen, Commissaire Européen, qui a fait des interventions publiques en faveur des logiciels libres.

On peut aussi mentionner l'existence d'une session « logiciels libre » lors de la conférence IST'2000 qui s'est tenue à Nice en novembre 2000. Parmi les projets européens en cours, citons le programme IDA (*Interchange of Data between Administrations*) et, dans le domaine de la santé, les projets SPIRIT (inventaire et dissémination de logiciels libres) et SMARTIE (logiciels libres pour des systèmes experts).

Au niveau des états membres, des initiatives nationales ont également été lancées. Par exemple, en France, le Premier Ministre a chargé le député T. Carcenac d'une mission sur l'analyse des systèmes d'information des administrations, examinant notamment l'intérêt des logiciels libres (voir le rapport sur http://www.internet.gouv.fr/carcenac.htm). Des recommandations similaires ont été émises pour les administrations allemandes. De plus, le Ministère allemand de l'Economie recommande l'utilisation de logiciels libres pour les PME et PMI.

Sur le plan juridique, les licences GPL ont un fondement légal en Europe, mais il n'y a aucune jurisprudence en la matière. Des problèmes posés concernent la protection des consommateurs (détermination des responsabilités en cas de dommages dus à un logiciel libre).

[2] L'adhésion du GT *RIS* à leur liste électronique *freesw* pourrait être pertinente.

Tullio Vardanega (ESA)

Le *Technical & Operational Support* (TOS) de l'ESA regroupe des départements offrant un soutien technique pour les applications spatiales, infrastructure sol ou embarqué. La représentation de l'ESA au sein du GT LL se situe côté embarqué.

L'ESA a un rôle actif de promoteur des logiciels libres dans ce domaine. Un séminaire a déjà été organisé sur le sujet : ftp://ftp.estec.esa.nl/pub/ws/opensource/OpenSourceSeminar.htm. Les conclusions en sont les suivantes. On trouve des avantages certains aux logiciels libres : interopérabilité, barrière d'entrée plus faible pour accéder à la technologie. Par contre, des problèmes ont été évoqués. L'introduction de logiciels libres dans des projets suppose une volonté de coopération qui est contraire à la culture industrielle (conflit avec la protection des intérêts privés). Un apport initial (fonds publics ?) est nécessaire pour démarrer de tels projets. Enfin, des évolutions dans les processus industriels devront être provoquées. Notamment, le coût va migrer du développement de produits vers la qualité de service.

Jean-Michel Tanneau (THALES)

THALES Technologies et Méthodes a lancé un groupe de travail « logiciels libres » en septembre 1999. Les objectifs étaient de recenser les logiciels libres utilisés dans l'entreprise, et d'identifier les risques liés à ces utilisations. En particulier, un effort important a été mis sur l'analyse des licences sur le plan juridique. Les travaux du groupe ont débouché sur la mise en place de recommandations et de procédures internes cadrant l'utilisation de logiciels libres. Une action de sensibilisation à ces recommandations a été menée auprès des chefs de projet et développeurs. Les informations recueillies en recensant les logiciels libres sont en cours de capitalisation dans une base de données (qui répertorie aussi bien des COTS que des LL).

Sur le plan technique, les problèmes posés concernent la sélection/évaluation de logiciels libres, la gestion de configuration des versions, et la maintenance à long terme (typiquement 15-20 ans). THALES est demandeur de partenariats industriels sur ces problèmes, et développe ses propres compétences internes. Notamment, un réseau d'excellence Linux a été créé. Une utilisation dans le cadre d'applications temps-réel embarquées est envisagée, mais les solutions techniques restent à définir (modification du noyau, sur-couche, ou évolution d'un OS vers Linux). Par ailleurs, pour quelques développements internes, THALES a expérimenté un nouveau modèle de développement coopératif, similaire au processus de création de logiciels libres.

En conclusion, le groupe de travail au sein de THALES a d'abord mis l'accent sur l'aspect réduction des risques. Il s'agit maintenant d'évaluer les avantages offerts par les logiciels libres, en analysant les aspects fiabilité, couverture de besoins fonctionnels,…

Philippe Coupoux (Technicatome)

Les systèmes développés à Technicatome se caractérisent par des petites séries, avec une longue durée de vie. Par exemple, dans le cas de la propulsion navale, les bateaux sont en service pendant 40 ans ; les systèmes bord sont installés pour au moins 10 ans, et posent des problèmes de maintenance (obsolescence, vieillissement,…). Ces systèmes bord ont de fortes contraintes de disponibilité (fonctionnement 24H/24 pendant plusieurs mois). Les contraintes de sûreté varient, allant du non classé au niveau le plus critique.

Pour des raisons de coût, une tendance lourde à Technicatome est l'abandon de solutions internes. On s'oriente donc vers l'utilisation de COTS ou de logiciels libres. Ces derniers offrent des avantages qui peuvent les faire préférer aux COTS : maîtrise des évolutions selon le calendrier projet plutôt que selon un calendrier imposé par le fournisseur, pas d'arrêt de commercialisation et de maintenance, migration sur différents systèmes facilitée par l'accès aux sources. Par contre, certains inconvénients sont à souligner : pas de support unique identifié, gestion de configuration à faire, authentification de l'origine des versions du LL, caractérisation de la « qualité » d'une version. Un problème commun aux COTS et aux LL concerne les exigences de certification : l'utilisation de ces composants est proscrite aux niveaux de criticité les plus hauts. Les besoins de Technicatome iraient jusqu'à des systèmes de niveau B, ce qui paraît actuellement difficile.

Corentin Desquilbet (CEAT)

Le CEAT est un centre d'expertise et d'essais de la Délégation Générale pour l'Armement (DGA). Il joue notamment un rôle de représentant des autorités de certification. Il est impliqué dans la certification de systèmes informatiques du secteur civil (40%) et militaire (60%), essentiellement dans le domaine aéronautique, mais aussi dans le domaine spatial, et, de façon plus marginale, dans les domaines nucléaire et des transports terrestres. La participation au Groupe de Travail du *RIS* est vue comme un moyen d'influencer les donneurs d'ordre et fournisseurs, en amont des projets soumis à certification.

Dans le passé, un COTS (LynxOS) a été accepté pour un système de niveau C. Mais les certifieurs ont exigé que l'intégrateur ait accès au code source, et ce composant a été soumis aux mêmes règles que les autres (couverture structurelle par test,…). A noter que dans les documents normatifs du domaine avionique, on n'emploie pas le terme de COTS, mais celui de PDS (Previously Developed Software).

On arrive aux limites du DO-178B qui, n'ayant pas évolué depuis 1992, conserve une logique verticale : on certifie des avions. Dans un futur proche, il est envisagé de pouvoir donner des certificats à des composants logiciels. Dans ce contexte, l'ouverture du code source sera nécessaire.

Philippe David, Eric Conquet (Astrium)

Actuellement, Astrium a peu d'utilisations explicites de logiciels libres sur des projets opérationnels spatiaux, par contre il y a des utilisations cachées dans les environnements de développement (gcc, …).

L'utilisation de logiciels libres est vécue comme une tendance lourde, poussée par les donneurs d'ordre. Deux exemples d'initiatives de l'ESA sont mentionnés :

- le projet FRESCO (FRee Software for ERC32 Systems C0operation), qui vise à la mise en place d'un consortium pour maintenir des logiciels libres basés sur SPARC ;
- la mise à disposition des sources VHDL du LEON (processeur SPARC spatial de nouvelle génération).

Pour ce deuxième exemple, on a constaté une position attentiste des industriels du domaine (dont Astrium), qui sont restés dans une logique d'observation plutôt que de contribution.

Les avantages des logiciels libres sont multiples. Ils implémentent généralement des standards, et permettent de déverrouiller le marché vers des entreprises qui accèdent ainsi plus facilement à la technologie. Du point de vue de la validation, ces logiciels ont non seulement un grand nombre d'utilisateurs, mais également un grand nombre de développeurs et de dévermineurs. Des ensembles de tests sont parfois livrés avec le code source. Par contre, se pose le problème de la gestion des évolutions, et de la conception d'architectures robustes à ces évolutions.

Les questions que se pose Astrium concernent les formes d'intervention possibles dans le « cercle vertueux » des logiciels libres (processus de création, appropriation par les utilisateurs qui deviennent des contributeurs). En particulier, quelles actions pourraient être menées pour guider les évolutions des produits et leur validation ? Comment organiser l'entreprise pour rentrer dans le « cercle vertueux » ?

3. Compétences du Groupe et thèmes à traiter

Le Groupe de Travail aurait besoin d'être renforcé sur deux points : 1) il n'y a pas de représentant de la communauté « logiciels libres » (contributeurs) ; 2) sur les aspects juridiques, THALES est le plus avancé, les autres industriels n'ayant pas encore entrepris de réflexions formelles dans ce sens. Philippe David et Hélène Waeselynck rappellent que la convention du *RIS* permet de faire intervenir occasionnellement des orateurs extérieurs, identifiés par l'intérêt et la proximité de leurs travaux avec ceux menés dans le Groupe. Les personnes ou organismes suivants sont mentionnés comme susceptibles d'être contactés :

- contributeurs aux logiciels libres : Franco Gasperoni (ACT Europe, pour GNAT Ada 95), Brian Bray (Minoru Development Corporation, logiciels libres dans le domaine de la santé), Matthieu Herrb (LAAS-CNRS, maintenance de OpenBSD), Linux User Group régional.
- Experts sur les aspects juridiques : AFUL (Association Francophone des Utilisateurs de Linux et des Logiciels Libres), + chaque participant du GT LL contactera le service juridique de son entreprise.

Les aspects juridiques étant considérés comme un thème prioritaire, THALES présentera une brève synthèse de ces analyses lors de la prochaine réunion du GT.

La discussion s'engage sur certains des autres thèmes évoqués lors des précédentes interventions.

Les industriels dans le « cercle vertueux »

Quelques précisions sont apportées sur la notion de cercle vertueux. Dans le modèle idéal, il n'y a pas de rôle fixe (développeur, dévermineur, utilisateur). L'important est le rebouclage, qui permet de partager la connaissance. En pratique, le développement de logiciels libres est rarement anarchique : une communauté se structure et met en place une hiérarchie informelle (un *benevolent dictator* et ses *lieutenants*). Les contributeurs sont souvent des administrateurs systèmes, qui développent des logiciels libres pour répondre à leurs besoins propres.

Les avis sont partagés quant à la possibilité, pour les industriels, d'intéresser des individus (« artistes ») au développement de leurs applications. Certains pensent qu'il est irréaliste de compter sur la communauté pour développer des composants spécifiques à leur métier. Il vaut mieux envisager la création de logiciels libres en collaboration avec d'autres industriels du domaine. D'autres, au contraire, pensent qu'il est possible d'influer sur le développement de nouvelles applications par appel à la communauté, éventuellement en injectant des financements. En particulier, la définition de standards, formats ouverts, etc. est un moyen de favoriser ce type de développement.

Logiciels libres et certification

Il paraît difficile de faire collaborer deux logiques bien différentes, la culture du « libre » et le formel de la certification. Quelques suggestions sont cependant émises pour introduire plus de rigueur dans le processus de développement des

logiciels libres. On peut envisager de mettre des outils de développement et de validation à la disposition de la communauté, ainsi qu'un support de type base de données pour recenser les *bugs*[3]. Du point de vue de la validation, l'importance de disposer d'une spécification est soulignée. Pour cela, une piste est l'utilisation de formalismes comme ADL (*Assertion Definition Language*), spécialisés dans la description comportementale d'interfaces (voir http://adl.opengroup.org).

Au final, ce sera le rôle des industriels d'identifier les logiciels libres les plus matures, et de mener les actions nécessaires en vue de la certification (amélioration du LL, validation). Certains pensent qu'il sera nécessaire de découper les logiciels libres en petits morceaux validables, qui pourraient alors constituer des briques de base certifiées.

En cas de dommages dus à un logiciel libre, il semble que ce soit l'intégrateur qui soit pénalement responsable.

4. Mise en place de la logique de travail

L'alternative suivante est proposée : une logique de réunions par thème (terminologie, aspects juridiques, validation, architectures,…) ou par domaine d'application (spatial, aéronautique, nucléaire,…). Après discussion, la logique par thèmes est préférée.

Ceci conduit à envisager un premier plan de réunions :

Réunion 2: terminologie, catégorisation des LL, attentes et craintes (identification des priorités), faire un point sur les impacts juridiques (THALES).

Réunion 3: architecture et validation des LL et des systèmes basés sur des LL.

Réunion 4: processus de création et des évolutions des LL, processus d'utilisation des LL, certification, protection intellectuelle.

Réunion 5: aspects juridiques.

Réunion 6: sécurité, autres points techniques.

Ce plan est bien sûr susceptible d'évoluer en fonction des travaux du Groupe. Les dates suivantes sont proposées : 11 ou 12 juillet pour la réunion 2, date à choisir dans la semaine du 10 au 14 septembre pour la réunion 3, date à choisir dans la semaine du 12 au 16 novembre pour la réunion 4.

5. Mode de fonctionnement

La discussion a porté sur la confidentialité des échanges au sein du Groupe de Travail. Chaque participant devra faire part de ses contraintes aux animateurs, notamment en ce qui concerne la diffusion des compte-rendus de réunion.

Le site web du Groupe de Travail sera organisé sur deux niveaux : un niveau public, et un niveau réservé aux membres du GT, accessible par mot de passe.

[3] Cela existe déjà pour Linux, voir http://hands.stanford.edu/linux/