

# Active Workspaces

## Campagne ADT 2015

### Récapitulatif

- **Titre & Acronyme :** Active Workspaces
- **Porteur de l'ADT :** Eric Badouel et Loïc Hélouët **Courriel :** eric.badouel@inria.fr  
**CRI :** Rennes Bretagne-Atlantique **EPI :** SUMO
- **Il s'agit d'une nouvelle ADT pour deux ans :** Oui
- **Les partenaires internes (EPI/CRI) et externes (autres labos, industriels) qui contribuent au travail décrit dans cette demande d'ADT :**
  - Christophe Morvan, Enseignant à l'Université Marne-la-Vallée, en délégation dans l'équipe SUMO.  
Email : christophe.morvan@inria.fr
  - **Equipe-projet ALOCO du LIRIMA**, Yaoundé, Cameroun.  
Contact : Georges-Edouard Kouamou  
Chargé de cours, Ecole Nationale Supérieure Polytechnique de Yaoundé.  
Email : georges.kouamou@lirima.org ou georges\_edouard@yahoo.com
  - **Centre Pasteur du Cameroun.**  
Collaboration dans le contexte de la thèse de Robert Nsaibirni qui vise à développer un système de surveillance épidémiologique basé sur le modèle des Active Workspaces.  
Contact : Gaëtan Texier, Chef du Service d'épidémiologie et de santé publique, Centre Pasteur du Cameroun  
Email : texier@pasteur-yaounde.org ou gaetex1@gmail.com  
Autres participants: Robert Nsaibirni Fondze Jr, PhD Student, Computer Sc., University of Yaoundé I  
Email : robert.nsaibirni@lirima.org ou nsairobby@gmail.com
  - Carlo Ferigato, **European Community Joint Research Center**, Ispra.  
Collaboration dans le contexte d'un projet de coordination de débats parlementaires sur la base des Robert's rule of order (RRO). L'objectif est d'implémenter cette coordination sous forme d'Active Workspaces.  
Contact : Carlo Ferigato, European Community Joint Research Center, 2749 via Enrico Fermi, 21047 Ispra, Italy  
Email : carlo.ferigato@jrc.ec.europa.eu
- **Budget :**

Les ressources nécessaires à la réalisation de ce projet de développement sont limitées à un poste de travail, à acheter en début de projet, et à des frais de formation et de

diffusion. Tout les logiciels utilisés (freeplane, JAVA,...) sont des logiciels libres ou gratuits.

Budget <b>demandé, hors personnel</b> , pour l'ADT	2015 (3 mois)	2016 (12 mois)	2017 (9 mois)	Total
Total budget (en k€)	3.5	3	0.5	7

**Total des ressources **demandées** pour l'ADT :**

3 KEuros

**Total des ressources **demandées** pour la 1ère année (12 mois) :**

3 KEuros

- **Résumé :** Le projet de ce développemet est de construire un outil permettant de définir des espaces de travail utilisateurs sous la forme de cartes heuristiques (essentiellement des arbres), servant à organiser les tâches en cours de l'utilisateur et les données qui s'y rattachent. Ces cartes sont paramétrisées par des grammaires attribuées gardées (GAG) dont les règles sémantiques permettent d'automatiser les actions récurrentes de l'utilisateur. Les cartes heuristiques ainsi étendues deviennent des documents structurés dont l'évolution est intentionnellement décrite par les règles sémantiques. Nous souhaitons développer cet outil à partir d'un logiciel de mind mapping (freeplane), qui est facilement extensible par des mécanismes de plugins et de scripts. Une seconde étape du développement consistera à adapter l'outil dans un contexte distribué. Une GAG peut en effet être distribuée ce qui permet de décrire des systèmes collaboratifs déployés sur une architecture asynchrone en établissant une communication entre les espaces de travail des utilisateurs. Enfin, l'outil comportera un module pour la spécification de GAGs et leur construction modulaire, incluant des techniques de tissage d'aspects.

## 1 Introduction

Le but de l'ADT est de développer un prototype de systèmes collaboratifs centré autour de la notion d'espace de travail utilisateur (*Active Workspace*) et issu de recherches conduites dans l'équipe SUMO. Il s'agit d'une action exploratoire dont les résultats permettront : (1) de démontrer la pertinence du modèle des grammaires attribuées gardées mis en avant par SUMO (ce prototype est indispensable pour diffuser ces résultats dans la communauté du *Business Process Management* où les modèles d'IBM et de ses partenaires académiques sont prédominants), (2) de réaliser des études de cas significatives au sein de l'équipe Aloco du Lirima (Inria Intern. Lab. en Afrique) et en collaboration avec Carlo Ferigato au JRC à Ispra, (3) de réaliser de futures expérimentations pour nos recherches sur le sujet, (4) de servir de vitrine pour nos compétences lors de discussions avec des partenaires industriels.

## 2 Contexte : état des lieux et positionnement **avant** l'ADT

### Contexte scientifique

Ce projet s'inscrit dans une démarche initiée par IBM <sup>1</sup> consistant à centrer la spécification d'un système à flot de tâches (*workflow*) sur les artefacts métiers (*Business Artifacts*). Ces derniers sont des documents qui recueillent au long de leur cycle de vie les informations concernant un dossier introduit dans le système.

Nous avons introduit <sup>2</sup> un modèle déclaratif d'artefacts, basé sur des grammaires attribuées gardées (GAG), dans lequel le cycle de vie d'un artefact est implicite : son évolution dépend dynamiquement de la valeur de certaines données et des actions de l'utilisateur. Le modèle est donc à la fois centré sur les utilisateurs et dirigé par les données. Les utilisateurs communiquent par échanges de messages et ne disposent pas de mémoire partagée, ce qui conduit à un schéma de distribution simple d'une GAG sur une architecture asynchrone.

En quelques mots le modèle des GAGs se présente comme suit. L'espace de travail d'un utilisateur est donné par un arbre de tâches conforme à une grammaire. Les catégories syntaxiques de la grammaire, ou sortes, correspondent aux différentes tâches du système. Une sorte (et donc tous les noeuds de cette sorte) possède des attributs dont les valeurs sont des données nécessaires à l'exécution de la tâche (attributs hérités) ou au contraire des informations produites au cours de l'exécution de la tâche (attributs synthétisés). Chaque production de la grammaire est une règle décrivant comment réduire la tâche, associée au symbole en sa partie gauche, en la liste des sous-tâches données par les symboles en sa partie droite. Les règles sémantiques servent essentiellement à faire le lien entre la tâche et ses sous-tâches en transmettant les informations (en entrée et en sortie) entre les différents attributs. Ces règles sémantiques décrivent le calcul de la valeur d'un attribut

---

<sup>1</sup> A. Nigam and N.S. Caswell. Business artifacts: An approach to operational specification. *IBM Syst. J.* 42:428-445, July 2003. & Richard Hull. Artifact-centric business process models: Brief survey of research results and challenges. OTM 2008, vol. 5332 of LNCS, pp. 259-274, Springer, 2013.

<sup>2</sup> Eric Badouel, Loïc Hélouët, Georges-Edouard Kouamou, Christophe Morvan. A grammatical Approach to Data-centric Case Management in a Distributed Collaborative Environment, SAC 2015, Salamanca (version complète : <http://hal.inria.fr/hal-00990007>)

en fonction des actions de l'utilisateur et/ou de la valeur d'autres attributs (à la manière d'un tableur). Elles ne sont pas nécessairement de nature purement fonctionnelle mais peuvent avoir des effets de bords (actions sémantiques) comme l'envoi de messages ou la modification du système de fichiers ou d'une base de donnée locale. Les productions de la grammaire sont gardées par des motifs (*patterns*) qui apparaissent dans les positions héritées en partie gauche. Une production est ainsi autorisée en un nœud de l'arbre si la valeur des attributs hérités du nœud correspondent aux motifs de la production. On peut exploiter la structure de l'arbre, donnée par l'association de sortes à ses nœuds, pour définir des scripts qui permettent d'automatiser certaines activités récurrentes de l'utilisateur. Contrairement à l'application d'une production, l'exécution d'un script ne modifie pas la structure de l'arbre mais génère des informations (tableaux de bords) et/ou des effets de bord. Ils peuvent néanmoins être vus comme des attributs (synthétisés) particuliers.

### Contexte technologique

Nous comptons construire ce prototype comme une extension au logiciel libre d'édition de cartes heuristiques Freeplane.<sup>3</sup> Dans ce contexte Freeplane est utilisé essentiellement pour l'édition interactive de structures arborescentes. Freeplane est habituellement utilisé pour dessiner des cartes heuristiques (*Mind maps*). Nous l'utiliserons pour dessiner et faire évoluer les artefacts d'un système collaboratif. Une grammaire attribuée gardée va être codée par un ensemble de greffons (*add-ons*), écrits dans le langage de script Groovy, afin de contraindre le développement de la carte pour qu'elle reste conforme à la grammaire tout en exécutant les règles sémantiques attachées aux modifications de l'arbre.

## 3 Objectifs de l'ADT

L'objectif de ce développement est d'implémenter notre modèle des grammaires attribuées gardées au dessus de l'outil freeplane. Cet objectif peut se décomposer comme suit :

1. Paramétrage de Freeplane par une grammaire attribuée gardée pour que le rendu graphique de la carte heuristique reflète les structures grammaticales et que le menu contextuel d'un nœud fasse apparaître les productions applicables et les scripts disponibles pour le nœud donné dans la configuration courante.
2. Extension de Freeplane pour ajouter des fonctionnalités (macros) à l'outil (e.g. filtrage d'une carte heuristique, exécution de séquences de tâches répétitives,...), et permettre la spécialisation d'un modèle de gestion de documents actif (décrit par une GAG) à l'aide de scripts définis par l'utilisateurs.
3. Implémentation de mécanismes de communication entre cartes heuristiques, conformément à une grammaire globale, en vue d'utiliser le prototype pour l'exécution de systèmes collaboratifs distribués.
4. implémentation de modules pour l'écriture de règles de productions de la grammaire, l'écriture de scripts, et la composition par aspects et la réutilisation de GAGs.

---

<sup>3</sup><http://www.freeplane.org/wiki/index.php>

## 4 Sortie : positionnement après l'ADT

Le prototype développé dans le cadre de cette ADT doit permettre de démontrer les capacités de ce modèle auprès de la communauté scientifique mais aussi auprès d'utilisateurs potentiels comme, par exemple, des entreprises de conseil faisant de la formation en Mind-mapping, ou plus généralement auprès de la communauté travaillant sur le développement de Freeplane ou d'applications autour de Freeplane. Nous souhaitons aussi utiliser cet outil comme vitrine de nos compétences lors de discussions avec des partenaires industriels intéressés par la gestion de l'information (Workflows, Business Processes,...).

Il nous permettra de développer des études de cas suffisamment significatives pour mettre en lumière les bénéfices et les limites du modèle et nous permettre de préciser certains aspects du modèle qui n'ont pas été jusque là précisément détaillées : (1) le couplage d'une GAG avec des services externes (par exemple des valeurs provenant d'une requête à une base de données ou d'un appel à un service web peuvent être utilisées pour alimenter certaines positions héritées et les valeurs produites en positions synthétisées peuvent en retour alimenter la base de données), (2) le développement d'une méthodologie de conception de GAG où l'accent sera mis sur les données plutôt que sur les processus (par exemple en se concentrant sur la structure des formulaires mis en jeu et sur les dépendances fonctionnelles entre les champs de ces différents formulaires), (3) la construction modulaire de GAG (approche à base de composants dans laquelle l'interface d'une GAG est décrite sous forme de services, tissage d'aspects, ...).

Les études de cas dans lesquelles nous sommes actuellement engagés (sur le papier pour l'instant) sont les deux suivantes :

1. Un système de surveillance épidémiologique. Ce travail est effectué dans le cadre de la thèse de Robert Nsaibirni co-encadrée par le Centre Pasteur du Cameroun. Les différentes phases d'un tel système sont traditionnellement modélisées sous forme d'arbres de tâches dont la structure est figée. Cette rigidité ainsi que la non-prise en compte des spécificités des différents acteurs (qui ont des compétences et des habitudes de travail différentes) fait que le comportement observé ne suit jamais parfaitement la spécification nominale prévue.

Dans notre cas la structure de l'arbre dépend à la fois des données reçues et des décisions des différents acteurs ce qui offre plus de souplesse et d'expressivité. Par ailleurs, nous souhaitons définir pour chaque intervenant un espace de travail minimal que l'utilisateur pourra faire évoluer de manière progressive en fonction de ses propres usages.

2. En collaboration avec Carlo Ferigato de l'JRC à Ispra, nous nous intéressons à un système pour l'amendement des documents parlementaires et l'organisation des débats suivant les règles d'ordonnancement de Robert <sup>4</sup> en utilisant les standards de documents judiciaires, législatifs et parlementaires mis en avant par Akoma Ntoso <sup>5</sup>

---

<sup>4</sup><http://www.robertsrules.com/>, <http://www.cs.uu.nl/groups/IS/archive/henry/RobertReport.pdf> et <http://www.robertsrules.org/rror-00.htm>

<sup>5</sup>The ICT in parliament of the UN and Africa e-parliaments site: [www.akomantoso.org](http://www.akomantoso.org)

et les outils open sources développés dans le cadre de Bungeni. <sup>6</sup>

D'autres études de cas sont envisagées en collaboration avec l'équipe ALOCO dont un système éditorial de revue électronique et un système de e-learning distribué. Dans ce dernier cas, les avantages par rapport aux solutions traditionnelles sont d'une part sa résilience en milieu dégradé (lorsque la connexion à Internet n'est pas constante) du fait que les utilisateurs travaillent essentiellement hors-ligne et que la communication entre leurs espaces de travail est effectuée quand la connexion Internet est disponible, et d'autre part parce que le caractère déclaratif des GAG n'impose pas d'ordre d'exécution des différentes tâches et qu'on obtient ainsi une plus grande souplesse dans la conception et dans l'exécution des contenus d'apprentissage.

Le projet de développement se décompose en trois parties. La première partie consistera à implémenter la gestion de l'espace de travail d'un utilisateur. La deuxième partie consistera à étendre cette implémentation pour réaliser des espaces de travail collaboratifs. La troisième partie devra fournir des outils pour l'édition de grammaires et des usages de l'utilisateur. Des expérimentations ainsi que des développements théoriques seront rendus possibles à l'issue de chacune d'entre elles.

**Espace de travail d'un utilisateur :** On représente l'espace de travail d'un utilisateur sous la forme d'un arbre (carte heuristique sous Freeplane) dont les nœuds sont associés à des tâches à réaliser. La configuration de l'espace de travail est obtenue par paramétrage de Freeplane par une grammaire attribuée gardée. Le menu contextuel d'un nœud associé à une sorte de la grammaire doit faire apparaître les productions autorisées et les scripts disponibles en ce nœud pour la configuration courante. L'utilisateur peut alors soit exécuter un script ou développer le nœud en lui appliquant une production autorisée. Le cas échéant une interface de saisie s'ouvre pour l'introduction des paramètres requis. Les autres nœuds peuvent être développés librement en utilisant les fonctionnalités de Freeplane (menu contextuel, raccourcis claviers ...). On introduira un peu de "sucre syntaxique" (1) pour les productions : par exemple afin d'utiliser des grammaires algébriques étendues (modalités "many", "some", "optional"), et (2) pour les scripts : afin d'offrir une syntaxe de plus haut niveau qu'on saura traduire en règles sémantiques (typiquement ces dernières peuvent contenir de nombreuses règles de copie qu'il peut être laborieux d'explicitier). De cette façon un espace de travail devient très largement configurable par l'utilisateur qui a ainsi la possibilité de raffiner progressivement la définition de son espace de travail en fonction de ses besoins. A ce stade, l'outil présentera le même genre d'usages qu'une carte heuristique usuelle (organiser ses données et ses tâches) avec des fonctionnalités supplémentaires permettant d'automatiser (ou d'automatiser partiellement) les activités récurrentes de l'utilisateur.

**Espace Collaboratif :** Un système collaboratif peut-être obtenu en assurant des communications entre les espaces de travail des différents intervenants. Plus précisément, un système collaboratif peut-être représenté par une grammaire attribuée gardée. L'espace

---

<sup>6</sup>Open source parliamentary and legislative applications of Africa i-Parliament: [www.bungeni.org](http://www.bungeni.org)

de travail d'un acteur correspond à une sous-grammaire du système associée à un sous-ensemble de sortes de la grammaire de départ. On dispose d'un mécanisme simple pour distribuer une GAG sur ses sous-grammaires obtenues par partition de la grammaire initiale. Cette distribution repose sur un mécanisme de souscriptions permettant à un site de souscrire à la valeur d'un attribut produit par un site distant. La valeur d'un attribut, lorsqu'elle est déterminée, est transmise aux sites y ayant souscrit. Cette valeur contient souvent elle-même de nouvelles souscriptions, ce qui permet d'établir des communications bi-directionnelles entre les différents sites. Dans la pratique nous ne souhaitons pas partir de la grammaire globale, qui restera implicite, mais plutôt décrire la communication entre les différents espaces de travail en incorporant le mécanisme de souscription dans les règles sémantiques.

**Outils pour la conception modulaire de GAG :** Freeplane possède des mécanismes de filtrage assez complets permettant de ne visualiser qu'une partie de la carte heuristique ce qui permet à l'utilisateur de se concentrer, à un moment donné, sur un aspect particulier des informations se trouvant dans la carte (en dehors bien sûr de la simple possibilité de pouvoir ouvrir ou refermer l'arborescence issu d'un nœud donné). Dans le cas des GAG on peut considérer qu'un aspect est donné par un sous-ensemble des attributs (deux aspects différents peuvent néanmoins avoir des attributs communs). On peut alors projeter un espace de travail sur un aspect particulier. Inversement, on peut utiliser cette notion d'aspects pour la réutilisation de spécifications : on a défini une opération de tissage de grammaires permettant de définir une nouvelle grammaire dont les grammaires de départ apparaissent comme des aspects particuliers de celle-ci (au sens précédent). La modularité peut aussi provenir d'un partitionnement des sortes (et non des attributs). Un composant correspond dans ce cas à un *rôle*. Comme décrit précédemment, cette opération est liée à la distribution des GAG et l'assemblage est assuré par un mécanisme de souscriptions. Il sera naturellement utile lorsqu'on compose des spécifications associées à des aspects ou à des rôles particuliers de renommer de façon adéquate certains éléments (sortes et/ou attributs) afin d'obtenir le résultat attendu. Nous avons ainsi besoin d'un langage de composition permettant d'explicitier ces renomages, tissages et souscriptions afin de pouvoir écrire des descriptions complexes de GAG à partir de composants simples.

## 5 Mise en œuvre prévisionnelle de l'ADT

Les personnes impliquées dans cette ADT seront

- Un ingénieur
- Eric Badouel et Loïc Hélouët (encadrants)
- Christophe Morvan, Georges-Edouard Kouamou, Gaétan Texier, Carlo Ferigato (Collaborateurs extérieurs)
- Robert Nsaibirni (Doctorant)
- Un ingénieur du SED (suivi du projet)

## 5.1 Identification des rôles et organisation

Le travail de développement sera principalement réalisé par l'ingénieur, avec un soutien des encadrants pour les choix techniques et l'architecture du logiciel. Les collaborations extérieures serviront à la réalisation de cas d'étude, et utiliseront le prototype pour valider les différentes étapes de la conception.

## 5.2 Planification prévisionnelle

Le projet de développement est décomposé en 7 tâches principales, elle mêmes déclinées en sous-tâches. Pour chaque tâche, nous définissons les participants, le temps de réalisation nécessaire, les livrables attendus.

**Tâche 0 : Apprentissage de l'existant :** Cette tâche consistera à acquérir les connaissances techniques nécessaires à la réalisation du projet : freeplane, GAG, ....

- T0.1 : apprentissage de Freeplane (0.5 hm, Ingénieur).
- T0.2 : apprentissage du modèle des GAG (1 hm, Ingénieur).
- T0.3 : apprentissage du langage de script Groovy et de l'environnement de programmation autour de Freeplane (1 hm, Ingénieur).

Livable:

L0.3 : Document de synthèse sur les mécanismes d'extension de Freeplane.

**Tâche 1 : Paramétrage de Freeplane pour définir un espace de travail utilisateur:**

- T1.1 : Analyse et rédaction de spécifications (1.5 hm, Ingénieur).
- T1.2 : Implémentation des règles sémantique dans Freeplane (2 hm, Ingénieur).
- T1.3 : Ajout d'interfaces (scripts, menus,...) à Freeplane pour en faire un outil de gestion d'espace documentaire (1 hm, Ingénieur).

Livrables:

- L1.1 : Document de spécification : fonctions, architecture du code, structures de données.
- L1.2 : Code : première version du prototype intégrant des GAG à Freeplane.
- L1.3 : Code : Extension du L1.2 avec des interfaces dédiées.
- L1.4 : Manuel d'utilisation.

**Tâche 2 : Extension des GAG pour une personnalisation utilisateur:**



- T2.1 : Mécanismes d'extension (6hm : E. Badouel, L. Hérouët, G.-E. Kouamou, C. Morvan, R. Nsaibirni, 1hm : Ingénieur).
- T2.2 : Implémentation de "sucre syntaxique" au dessus des règles des GAG (1 hm, Ingénieur).
- T2.3 : Implémentation de mécanismes de customisation utilisateur (scripts) (3 hm, Ingénieur).

Livrables:

- L2.1 : Document de définition des mécanismes de personnalisation.
- L2.2 : Code : extension du prototype intégrant des macro règles (sucre syntaxique).
- L2.3 : Code : extension du prototype intégrant un langage de script utilisateur.

**Tâche 3 : Distribution des espaces utilisateurs:**

- T3.1 : Etude des solutions et architectures distribuées existantes (0.5 hm Ingénieur).
- T3.2 : Réalisation d'un prototype distribué (4 hm, Ingénieur).

Livrables:

- L3.1 : Rapport d'étude.
- L3.2 : Code : extension du prototype intégrant distribution et communications entre sites.

**Tâche 4 : Conception des GAG**

- T4.1 : Mécanismes de conception modulaire de GAG: tissage , réutilisation,... (6hm : E. Badouel, L. Hérouët, G.-E. Kouamou, C. Morvan, R. Nsaibirni)
- T4.2 : Outils d'édition de GAGs, scripts, etc (6hm Ingénieur)

Livrables:

- L4.1 : Rapport de recherche.

**Tâche 5 : Etudes de cas**

- T5.1 : Modélisation d'un système de surveillance épidémiologique (6 hm E.Badouel, R. Nsaibirni, G. Texier).
- T5.2 : Modélisation des règles de Robert (6 hm : E.Badouel, C. Ferigato).

Livrables:

- L5.1 : Rapport : Modèle et retour d'expérience.

- L5.2 : Rapport : Modèle et retour d'expérience.

**Tache 6 : Communication et animation:** Diffusion de l'outil, participation à des forums

- T6.1 : Contribution à des forums autour de Freeplane (1 hm Ingénieur).
- T6.2 : Diffusion des résultats (Forums, réalisation de présentations) (2 hm Ingénieur).

## 6 Ressources

Les ressources techniques demandées sont limitées à un poste de travail pour l'ingénieur, à des frais de formation, et à des frais de déplacement pour formation ou diffusion.

### 6.1 Ressources humaines

Les ressources humaines utilisées dans le projet sont:

- Un ingénieur de développement (24hm)
- Le personnel pour l'encadrement de l'ingénieur (E. Badouel, L. Hélouët), et son suivi par un ingénieur conseil du SED (10% de son activité).

### 6.2 Aspects budgétaires

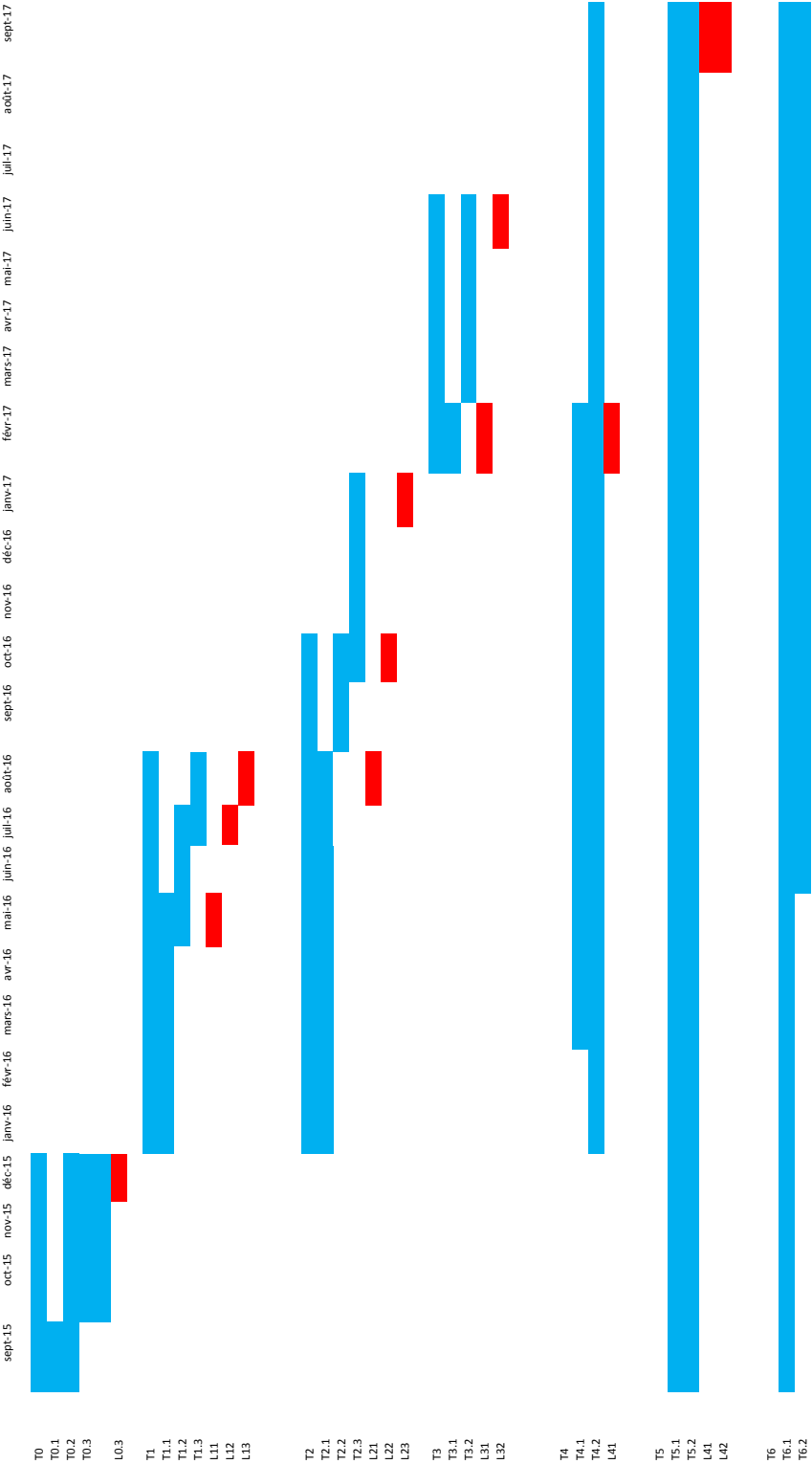
Les frais de matériel concernent le poste de travail de l'ingénieur. Nous prévoyons quelques frais de formation, des déplacements en France, pour ces formations ou pour contribuer à la dissémination des résultats de l'ADT.

Ressources <b>demandées</b> (en k€)	2015 (3 mois) versement 1 en sept.-oct.	2016 (12 mois) versement 2 et 3 en jan.-fév. et mai-juin	2017 (9 mois) versement 4 en jan.-fév.	Total
Missions	0.5	0.5	0	1
Animation	0	0.5	0.5	1
Matériel scientifique	3	0	0	3
Prestations externes	0	0	0	0
Autre (Formation)	0	2	0	2
<b>TOTAL</b>	<b>3.5</b>	<b>3</b>	<b>0.5</b>	<b>7</b>

## 7 Suivi et évaluation

Le pilotage de l'ADT se fera principalement en fonction des livrables. E. Badouel et L. Hélouët consacreront chacun une partie de leur temps à l'encadrement du développement (respectivement 10 et 5 %).

Le planning escompté est représenté page suivante. Il définit le délai de réalisation de chaque tâche, ainsi que les dates de livrables.



L'avancement et le succès du projet pourrait être évalué par les experts suivants:

- Expert 1 : Albert Benveniste (Inria Rennes, équipe Hycomes, Email : [albert.benveniste@inria.fr](mailto:albert.benveniste@inria.fr)). Albert Benveniste dirige l'action CominWeb, dont l'objectif est le développement d'espaces de recherche collaboratifs.
- Expert 2 : Marlon Dumas ( Univ. Tartu, <http://kodu.ut.ee/~dumas/>, Email : [marlon.dumas@ut.ee](mailto:marlon.dumas@ut.ee)).
- Expert 3 : François Charoy (Inria Lorraine, équipe Score, Email: [francois.charoy@inria.fr](mailto:francois.charoy@inria.fr)). François est extérieur au projet, mais débute cependant une collaboration avec le projet Aloco.

## Annexe A : Description des partenaires

- Le projet Aloco, pour *architectures logicielles à Composants*, fait partie du LIRIMA qui est l'Inria International Lab. en Afrique : <https://lirima.inria.fr/fr/>. Cette équipe est basée à l'Ecole nationale supérieure polytechnique (ENSP) de Yaoundé au Cameroun et est partenaire de l' EPI SUMO (Eric Badouel).
- Le Centre Pasteur du Cameroun est un établissement camerounais membre du Réseau international des Instituts Pasteur dont il partage la mission principale, la lutte contre les maladies infectieuses. Gaëtan Texier, Chef du Service d'épidémiologie et de santé publique au Centre Pasteur du Cameroun est médecin, membre de la faculté de médecine de Marseille. Robert Nsaibirni Fondze Jr. est doctorant à l'université de Yaoundé I. Il effectue ses travaux de thèse au Centre Pasteur du Cameroun, sous la co-direction de Gaëtan Texier, sur la modélisation et le développement d'outils informatiques pour les systèmes de surveillance épidémiologique.
- Carlo Ferigato est chercheur au Joint Research Center de la Communauté européenne à Ispra, en Italie <https://ec.europa.eu/jrc/en/about/jrc-site/ispra>. Il s'intéresse à l'amendement collaboratif des documents venant en support d'un processus délibératif (e.g., parlementaire) en liaison avec les règles d'ordonnancement des débats (Robert's rules of order). Le but est de formaliser ces règles pour aboutir à des outils de médiation dans le cadre de systèmes délibératifs pour aider à la prise de décision collective. Le but est aussi de rendre accessible à chaque intervenant, et à chaque moment du débat, les informations qui lui sont utiles et dont l'accès lui est autorisé. Un objectif secondaire est d'accroître la transparence des débats et des prises de décisions.