	<b>Document</b> : Spécifications d'Architecture Logicielle <b>Projet</b> : iFlybot 2012	<b>Réf.</b> : v0.4 <b>Version</b> : v1.0 <b>Date</b> : 20 mars 2012
	Projet ITIN / Université de Cergy-Pontoise	Page : 1 / 12

## Identification du document

**Titre du document** : Spécifications d'Architecture Logicielle


**Référence** : v0.4



**Version** : v1.0

**Date** : 20 mars 2012

**Nom du fichier** : SAL.docx

## Identification des intervenants dans le projet

	<b>Nom</b> : Philippe GAUSSIER <b>Nom de la société</b> : ETIS <b>Téléphone</b> : 01 30 73 66 10 <b>E-mail</b> : gaussier@ensea.fr	<b>Adresse</b> : Université de Cergy-Pontoise ETIS - UMR 8051 2 avenue Adolphe Chauvin 95302 Cergy Pontoise Cedex
<b>Contact 1</b> <b>Nom</b> : DESSALLE <b>Prénom</b> : Bruce <b>Téléphone</b> : 06 84 13 02 19 <b>E-mail</b> : bdessalle@cergy.itin.fr	<b>Contact 2</b> <b>Nom</b> : PELLISSON <b>Prénom</b> : Damien <b>Téléphone</b> : 06 89 88 29 14 <b>E-mail</b> : dpelliss@gmail.com	<b>Contact 3</b> <b>Nom</b> : NOIROT <b>Prénom</b> : Julien <b>Téléphone</b> : 06 84 00 35 13 <b>E-mail</b> : noirotj@gmail.com
<b>Contact 4</b> <b>Nom</b> : NESTY <b>Prénom</b> : Olivier <b>Téléphone</b> : 06 26 19 38 89 <b>E-mail</b> : nesty.olivier@gmail.com	<b>Contact 5</b> <b>Nom</b> : KAISER <b>Prénom</b> : Florian <b>Téléphone</b> : 06 83 70 51 74 <b>E-mail</b> : mail.de.florian.kaiser@gmail.com	

 	<p><b>Document</b> : Spécifications d'Architecture Logicielle  <b>Projet</b> : iFlybot 2012</p>	<p><b>Réf.</b> : v0.4  <b>Version</b> : v1.0  <b>Date</b> : 20 mars 2012</p>
	Projet ITIN / Université de Cergy-Pontoise	Page : 2 / 12

## Diffusion

Société	Destinataires	Nb
ETIS	Philippe GAUSSIER	1
ITIN	Bruce DESSALLE	1
Université Cergy Pontoise	Damien PELLISSON	1
Université Cergy Pontoise	Julien NOIROT	1
Université Cergy Pontoise	Olivier NESTY	1
Université Cergy Pontoise	Florian KAISER	1

 	<b>Document</b> : Spécifications d'Architecture Logicielle <b>Projet</b> : iFlybot 2012	<b>Réf.</b> : v0.4 <b>Version</b> : v1.0 <b>Date</b> : 20 mars 2012
	Projet ITIN / Université de Cergy-Pontoise	Page : 3 / 12

## Rédaction/Chaine de validation

	Nom	Qualité	Date
Rédigé par :	Damien PELLISSON	Responsable logiciel	11/03/2012
Approuvé par :	Bruce DESSALLE	Chef de projet	20/03/2012
Approuvé par :	Florian KAISER	Responsable Qualité	
Approuvé par :	Philippe GAUSSIER	Client	

 	<b>Document</b> : Spécifications d'Architecture Logicielle <b>Projet</b> : iFlybot 2012	<b>Réf.</b> : v0.4 <b>Version</b> : v1.0 <b>Date</b> : 20 mars 2012
	Projet ITIN / Université de Cergy-Pontoise	Page : 4 / 12

### Evolutions du Document

Version	Date	§ modifiés	Auteur
V0.1	18/01/2012	Squelette du document	Bruce DESSALLE
V0.2	11/03/2012	Spécifications sur l'architecture embarquée	Damien PELLISSON
V0.3	14/03/2012	Adaptation suite aux commentaires de Carina ROELS	Damien PELLISSON
V0.4	20/03/2012	Définition des protocoles utilisés	Olivier NESTY
V1.0	20/03/2012	Ajout information protocoles	Bruce DESSALLE

 	<p><b>Document</b> : Spécifications d'Architecture Logicielle  <b>Projet</b> : iFlybot 2012</p>	<p><b>Réf.</b> : v0.4  <b>Version</b> : v1.0  <b>Date</b> : 20 mars 2012</p>
	Projet ITIN / Université de Cergy-Pontoise	Page : 5 / 12

## Sommaire

---

1	Présentation du client .....	6
1.1	Activité du client.....	6
1.2	Besoins exprimés par le client.....	6
2	Objectifs du document .....	8
2.1	Portée du document .....	8
2.2	Références .....	8
3	Diagramme de déploiement.....	9
4	Protocole de communication .....	10
4.1	Entre le PC et la Gallop43 .....	10
4.2	Entre la Gallop43 et la RoboVero (ou X-FunPilot) .....	11

	<b>Document</b> : Spécifications d'Architecture Logicielle <b>Projet</b> : iFlybot 2012	<b>Réf.</b> : v0.4 <b>Version</b> : v1.0 <b>Date</b> : 20 mars 2012
	Projet ITIN / Université de Cergy-Pontoise	Page : 6 / 12

## 1 Présentation du client

### 1.1 Activité du client

Philippe GAUSSIER est le responsable du laboratoire ETIS, spécialiste en sciences de l'information et de la communication, il travaille sur la modélisation des mécanismes cognitifs impliqués dans la perception visuelle (vision pré-attentive et attentionnelle) et sur la modélisation de structures telles que l'hippocampe (pour des problèmes d'intégration d'informations spatio-temporelles).

Ces modèles servent de base pour imaginer des architectures de contrôle permettant à des robots mobiles d'apprendre à survivre dans un environnement à priori inconnu.

Ces robots utilisent généralement la vision comme source principale d'information ce qui met en avant l'importance du traitement de l'image.

Les tâches étudiées sont : l'apprentissage de conditionnements visio-moteurs, la reconnaissance de lieux (retour au nid), la construction et l'utilisation de cartes cognitives et enfin l'apprentissage de la discrimination d'objets et/ou de lieux.

### 1.2 Besoins exprimés par le client

L'objectif du client est de concevoir un drone pouvant se déplacer de façon entièrement autonome. Pour cela, il est mis à notre disposition deux plateformes pour effectuer nos tests, à savoir :

- Un modèle réduit de quadricoptère très maniable, simple d'accès et plus facile à mettre en œuvre (possibilité de tests en intérieur, en revanche plus sensible aux intempéries)
- Un modèle réduit d'hélicoptère bien plus puissant, pouvant embarquer plusieurs kilos de matériels et donc plus difficile à utiliser pour des raisons de sécurité

La plateforme 'quadricoptère' est considérée comme la solution prioritaire et constitue le produit final qui sera présenté au client au terme du projet. Au contraire, nous considérons la plateforme 'hélicoptère' comme un objectif bonus ou secondaire qui sera avancé au fur et à mesure en fonction de l'avancement des développements sur l'autre plateforme. Le fait de disposer de deux supports pour nos tests nous permettra de paralléliser certaines phases de conception et éventuellement de sous-traiter à des étudiants de M1 la partie hélicoptère. Les cartes de développement, également commandées en double, pourront nous permettre de répartir les tâches de développement en deux groupes.

 	<b>Document</b> : Spécifications d'Architecture Logicielle <b>Projet</b> : iFlybot 2012	<b>Réf.</b> : v0.4 <b>Version</b> : v1.0 <b>Date</b> : 20 mars 2012
	Projet ITIN / Université de Cergy-Pontoise	Page : 7 / 12

Les applications possibles du drone peuvent être très diverses :

- Exploration de zones difficilement accessibles
  - Ex : Entretien matériel et vérifications sur les avions, les buildings, les ponts etc.
- Surveillance d'une zone et reconnaissance des lieux et objets
  - Recherche de personnes disparues (avalanches, naufrages, catastrophes naturelles)
  - Reconnaissance morphologique de personnes (personnes recherchées, terroristes)
  - Détection d'intrusion, d'incendie de forêt

Il est essentiel et c'est l'un des buts finaux du projet d'avoir un vol en autonomie au moment où l'on vient à perdre le signal radio avec le drone. Ce dernier devra effectuer sa mission avant de revenir au lieu de départ.

Le projet est évidemment complexe et présente beaucoup de possibilités. Le client serait très satisfait d'obtenir à la fin du projet au moins une plateforme de développement opérationnelle et un système en boucle fermée sur l'hélicoptère qui lui permettrait des déplacements simples en toute autonomie.

	<b>Document</b> : Spécifications d'Architecture Logicielle <b>Projet</b> : iFlybot 2012	<b>Réf.</b> : v0.4 <b>Version</b> : v1.0 <b>Date</b> : 20 mars 2012
	Projet ITIN / Université de Cergy-Pontoise	Page : 8 / 12

## 2 Objectifs du document

Ce document a pour but de décrire les architectures logicielles utilisées au cours de la phase de conception du projet, à savoir : les IDE, les langages de programmation et les systèmes d'exploitation utilisés pour répondre au mieux aux besoins du client. Ce document détaille donc comment va être réalisé le projet iFlyBot 2012.

### 2.1 Portée du document

Ce document est lié au projet iFlyBot 2012 et rédigé d'après les spécifications d'exigences fonctionnelles ainsi que la note de cadrage.

### 2.2 Références

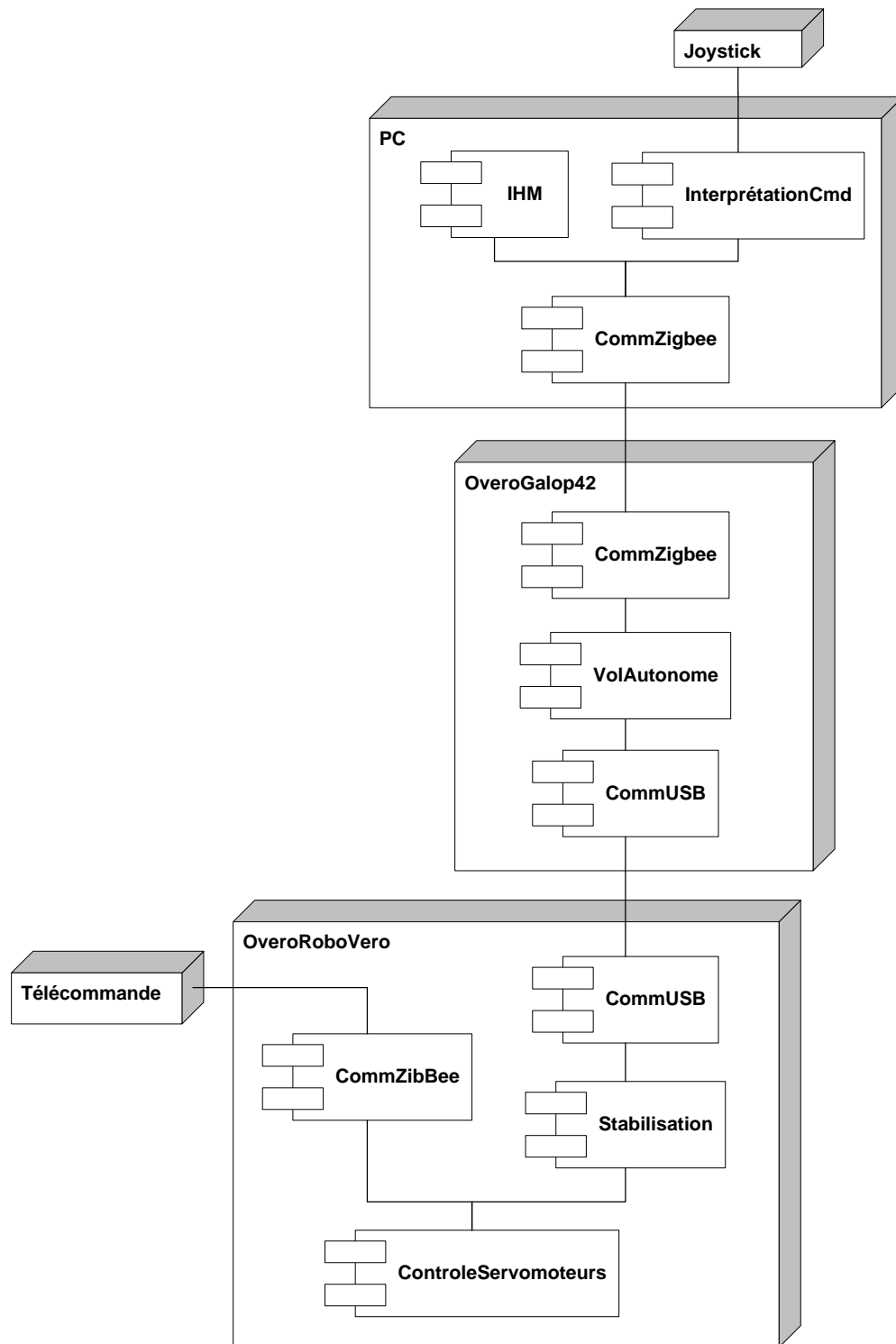
La table ci-dessous indique les documents disponibles pour les spécifications d'architecture logicielle :

Document	Notes
Spécifications d'exigences fonctionnelles	
Note de cadrage	
Planning du projet	
Spécifications IHM	



 	<p><b>Document</b> : Spécifications d'Architecture Logicielle  <b>Projet</b> : iFlybot 2012</p>	<p><b>Réf.</b> : v0.4  <b>Version</b> : v1.0  <b>Date</b> : 20 mars 2012</p>
	<p>Projet ITIN / Université de Cergy-Pontoise</p>	<p>Page : 9 / 12</p>

### 3 Diagramme de déploiement



## 4 Protocole de communication

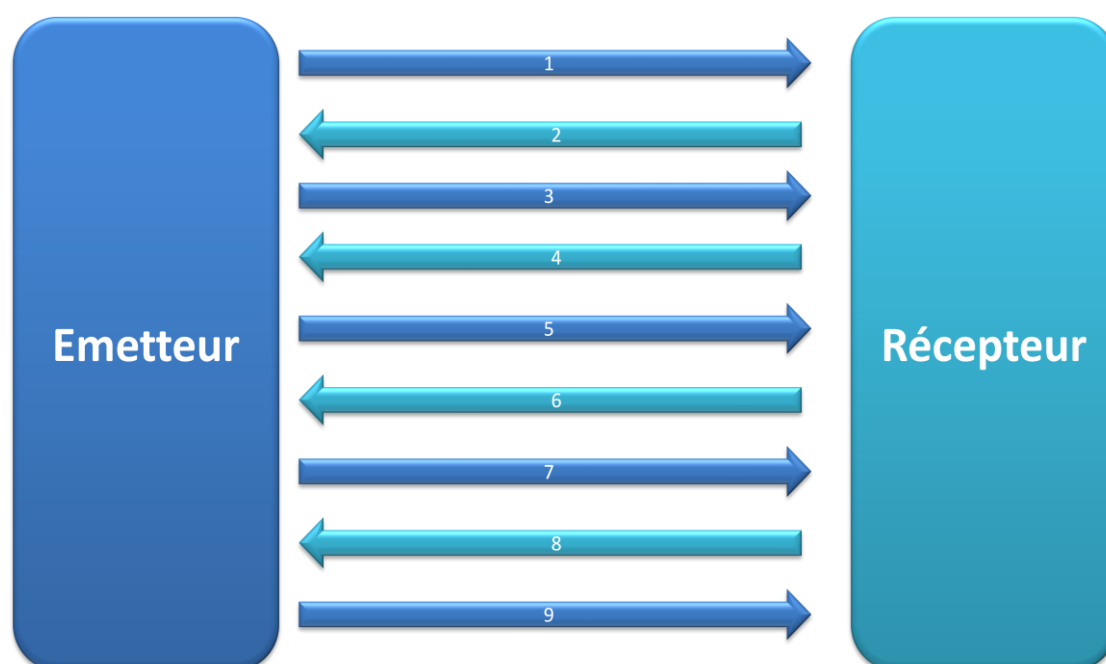
### 4.1 Entre le PC et la Gallop43

Le dialogue entre le PC et la Gallop43 se fait grâce à deux modules Xbee (l'un sur le PC et l'autre embarqué sur le drone). Ces deux modules communiquent via le protocole Zigbee. En effet, nous avons choisi ce protocole car il offrait le meilleur rapport portée/consommation.

Voici un tableau comparatif des protocoles de communication sans fils les plus fréquemment utilisés :

Protocole	Zigbee	Bluetooth	Wi-Fi
IEEE	802.15.4	802.15.1	802.11a/b/g/n/n-draft
Besoins mémoire	4-32 Kb	250 Kb +	1 Mb +
Autonomie avec pile	Années	Mois	Heures
Nombre de nœuds	65 000+	7	32
Vitesse de transfert	250 Kb/s	1 Mb/s	11-54-108-320 Mb/s
Portée	100 m	50 m	300 m

Ce protocole est basé sur la norme IEEE 802.15.4.



1. **Requête d'association** : demande d'association avec un coordinateur PAN (Personal Area Network).

	<b>Document</b> : Spécifications d'Architecture Logicielle <b>Projet</b> : iFlybot 2012	<b>Réf.</b> : v0.4 <b>Version</b> : v1.0 <b>Date</b> : 20 mars 2012
	Projet ITIN / Université de Cergy-Pontoise	Page : 11 / 12

2. **Réponse d'association** : réponse d'un coordinateur contenant l'état d'association (les différents états possibles incluent le succès de l'association, PAN à la capacité et accès refusé).
3. **Notification de dissociation** : utilisée par un périphérique ou un coordinateur pour informer d'autres nœuds de la dissociation.
4. **Requête de données** : utilisée pour demander des données d'un coordinateur.
5. **Notification de conflit d'ID de PAN** : transmise lorsqu'un conflit d'identificateur PAN est détecté.
6. **Notification orpheline** : utilisée par un matériel associé ayant perdu la synchronisation avec son coordinateur.
7. **Requête de balise** : utilisée pour la synchronisation et pour la transmission des informations de super trame.
8. **Réalignement du coordinateur** : utilisé par le coordinateur pour répondre à une commande de notification orpheline. Il est également utilisé lorsque les attributs PAN sont modifiés avec les informations de voie logique. Il peut être transmis à l'ensemble du PAN ou à un seul matériel orphelin.
9. **Requête GTS** : utilisée par un matériel associé pour demander l'allocation d'un nouvel emplacement en temps garanti (GTS) ou pour demander la désallocation d'un GTS existant à partir d'un coordinateur PAN. Elle définit également des champs GTS tels que la longueur, la direction et le type.

## 4.2 Entre la Gallop43 et la RoboVero (ou X-FunPilot)

Le protocole de communication utilisé entre la Gallop43 et la RoboVero est le protocole RS232.

Les messages utilisés dans les protocoles de communication sont les suivants,

	<b>Document</b> : Spécifications d'Architecture Logicielle <b>Projet</b> : iFlybot 2012	<b>Réf.</b> : v0.4 <b>Version</b> : v1.0 <b>Date</b> : 20 mars 2012
	Projet ITIN / Université de Cergy-Pontoise	Page : 12 / 12

Commande	Caractère envoyé	Valeur de retour	Caractère reçu	Commentaires
<b>Envoi d'ordres au plateau cyclique</b>	P(XXX,XXX,XXX,XXX)	Accusé de reception	ACK	<p>La RoboVero gère l'équilibre au niveau du drone, donc les valeurs finales appliquées aux servo-moteurs mais cette commande permet de définir l'avancement à appliquer selon la direction en termes de vitesse (valeur XXX comprise entre 0 et 255 où 255 correspond à la vitesse maximale applicable au véhicule).</p> <p>Les directions sont données dans le sens des aiguilles d'une montre, à savoir : Avant, Droite, Arrière, Gauche</p>
<b>Vitesse ascensionnelle</b>	A(XXX)	Accusé de reception	ACK	Où XXX représente la valeur de vitesse ascensionnelle. On suppose que la valeur 255 correspond à la vitesse positive maximum et 0 à la vitesse négative maximum
<b>Demande l'état du plateau cyclique</b>	GetP	Retourne l'état du plateau	P(XXX,XXX,XXX,XXX)	Format identique à l'envoi de la commande
<b>Demande l'état de la vitesse ascensionnelle</b>	GetA	Retourne l'état de la vitesse ascensionnelle	A(XXX)	Format identique à l'envoi de la commande
<b>Demande la valeur des accéléromètres</b>	GetB	Retourne l'état des accéléromètres	BX(XXX,YYY,ZZZ), BY(XXX,YYY,ZZZ), BZ(XXX,YYY,ZZZ)	Donne l'état des accéléromètres sur les 3 axes.