Coconode

Projet de configuration, d'automatisation et d'analyse de simulations sur des réseaux de capteurs

Noé-Jean Caramelli,
Minh Quan Ho,
Florian Lévêque

March 21, 2013

Plan

Présentation de l'équipe, responsabilités par type au début

	Minh Quan Ho	Florian Lévêque	Noé-Jean Caramelli
Chef de projet / Scrum Master			✔
Testeur	✔		
Ergonome		✔	
Architecte			✔
Développeur	✔	✔	✔

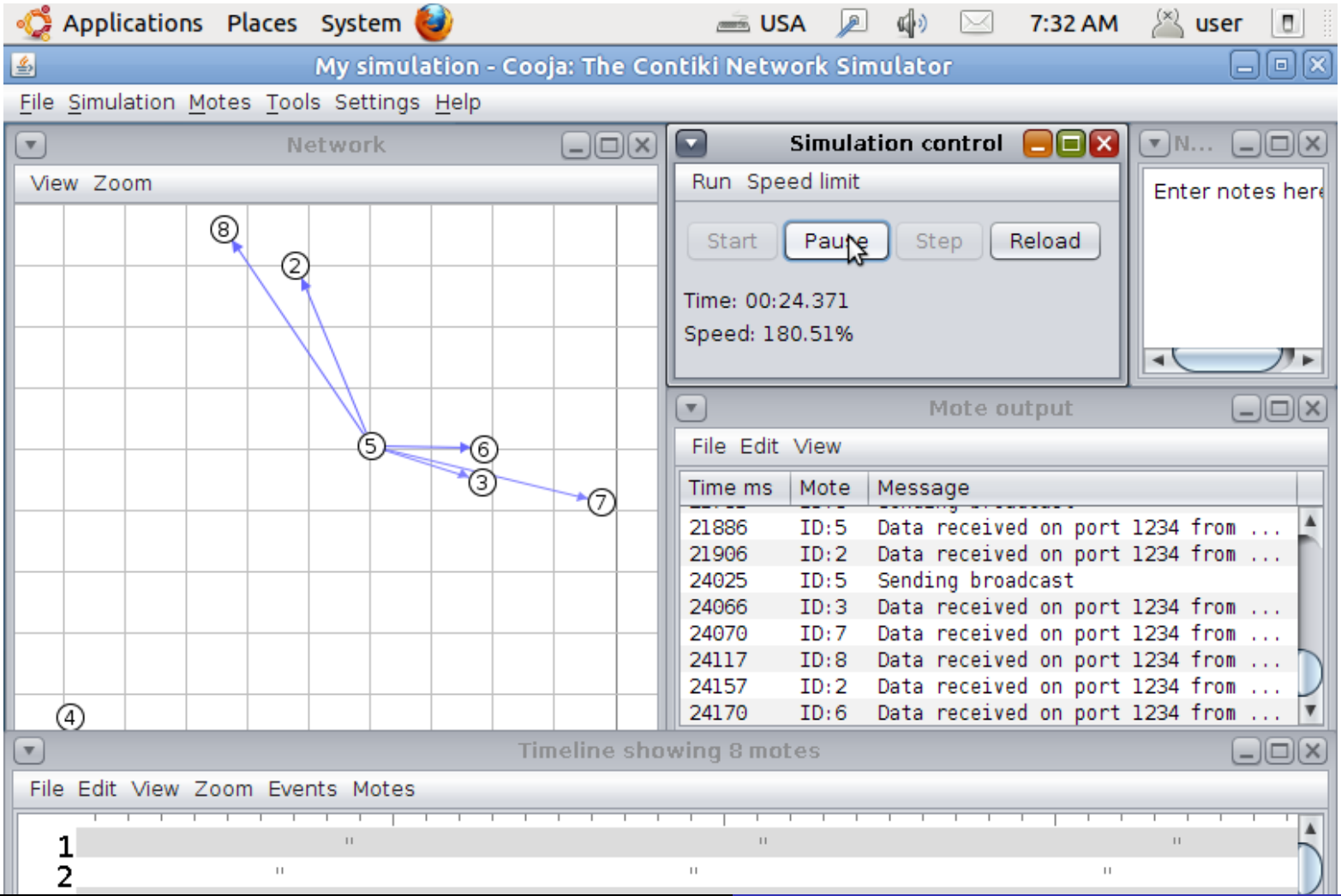
Présentation

- Réseaux de capteurs
- Pourquoi la simulation ?
  - Rapide
  - Peu coûteux
  - Paramétrable
  - Reproductible
- Désavantages
  - Pas représentatif de la réalité
  - Paramètres oubliés

Contiki

- Système d'exploitation
- Développé par le Swedish Institute of Computer Science (SICS)
  - Ultra léger
  - Flexible
- Plateforme d'émulation et de simulation

Cooja GUI


Applications Places System
USA
7:32 AM
user
My simulation - Cooja: The Contiki Network Simulator
File Simulation Motes Tools Settings Help
Network
View Zoom
Simulation control
Run Speed limit
Start
Pause
Step
Reload
Time: 00:24.371
Speed: 180.51%
Enter notes here
Mote output
File Edit View
Time ms
Mote
Message
21886 ID:5 Data received on port 1234 from ...
21906 ID:2 Data received on port 1234 from ...
24025 ID:5 Sending broadcast
24066 ID:3 Data received on port 1234 from ...
24070 ID:7 Data received on port 1234 from ...
24117 ID:8 Data received on port 1234 from ...
24157 ID:2 Data received on port 1234 from ...
24170 ID:6 Data received on port 1234 from ...
Timeline showing 8 motes
File Edit View Zoom Events Motes

Cooja

- Propriétés
  - Émulateur de Contiki
  - Communications entre les nœuds
  - Log de tous les évènements
- Inconvénients
  1. Pas d'interprétations des résultats
  2. Pas de répétitions des simulations
  3. Création des topologies manuelle
  4. Pas de planifications

Présentation du projet

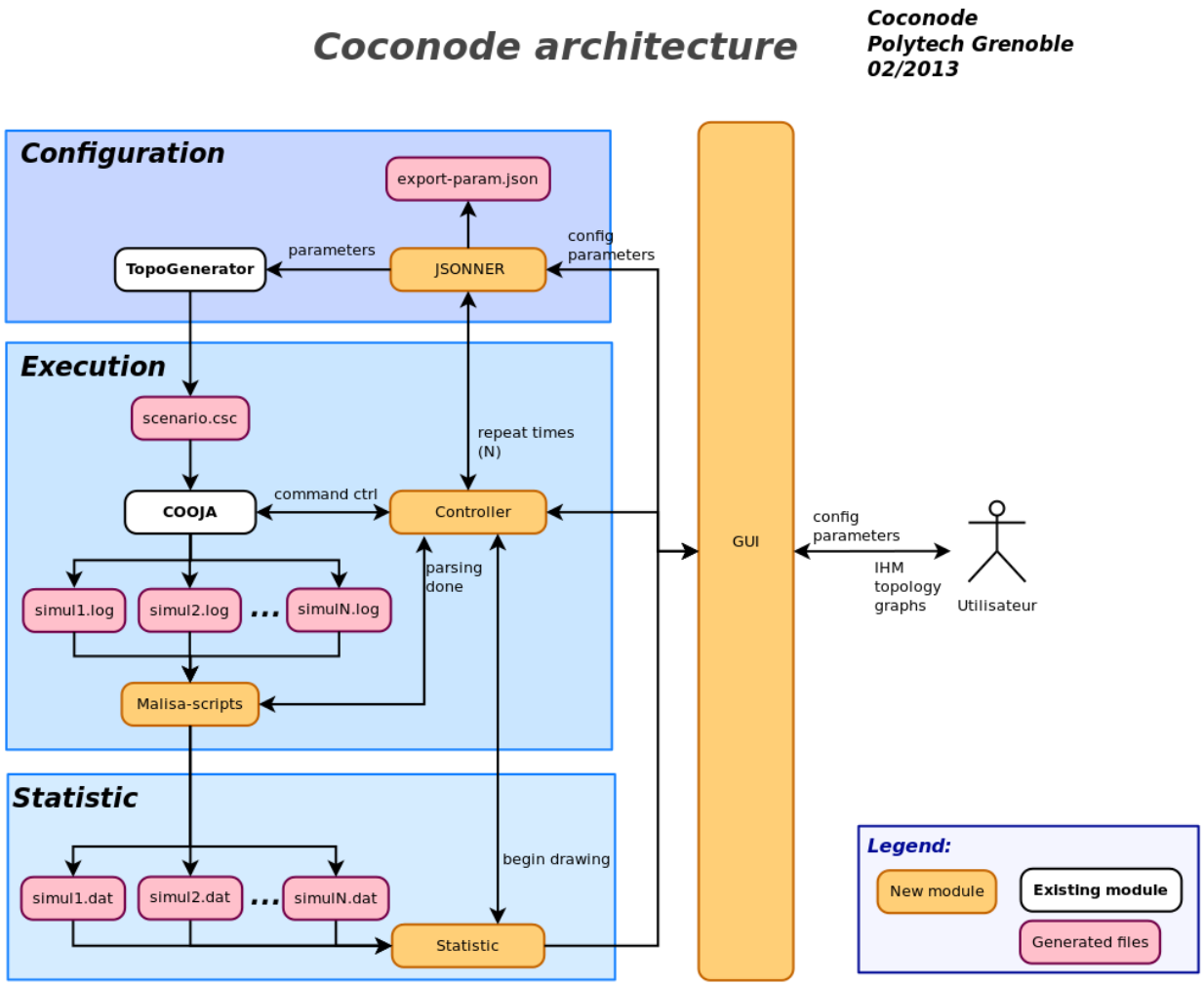
Interface graphique simple

- Configuration complète de simulations
- Planification (nombre de simulations, délai de lancement)
- Génération automatique d'une topologie
- Export / import des paramètres en JSON
- Monitoring de l'avancement et contrôles
- Génération de graphes statistiques

Aspects techniques

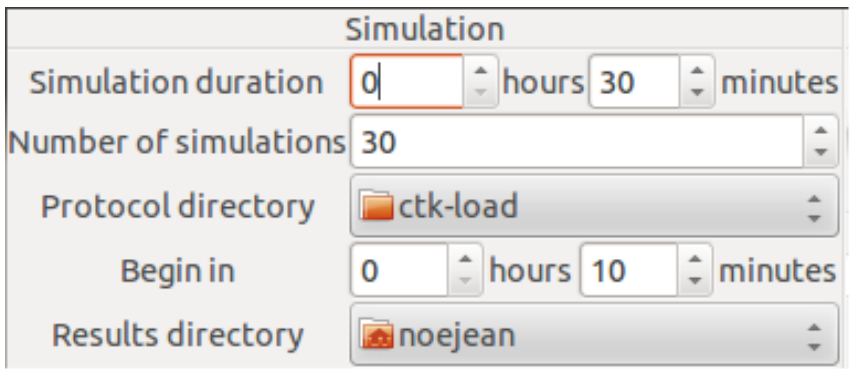
- C et bibliothèque graphique GTK (et gestion des threads)
- Bibliothèques dynamiques
- Compilation du code utilisateur pour la topologie
- Basé sur le noyau Cooja avec Java Native Interface (JNI)
- Gnuplot
- Json-glib

Architecture



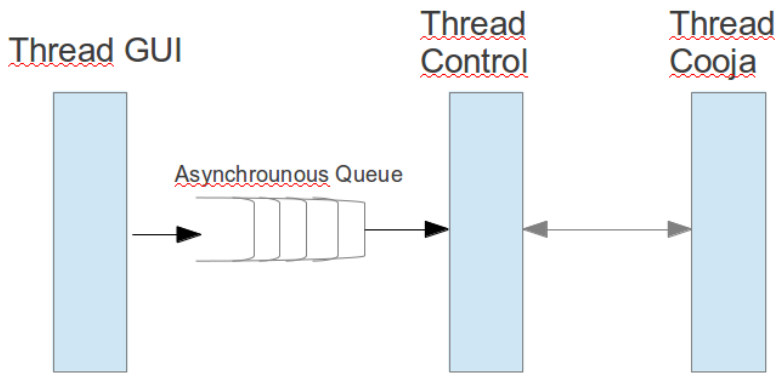
Configuration

- Durée de simulation
- Simulations planifiées
- Taille de l'échantillon
- Choix du protocole de routage



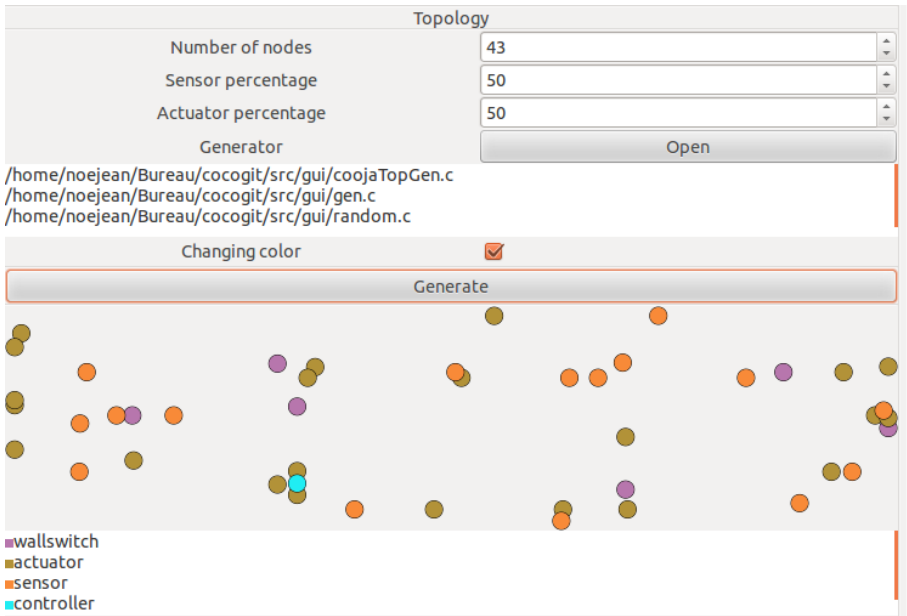
Scheduler

- Plannifier le lancement des simulations
- File d'attente de simulations + un ordonnanceur



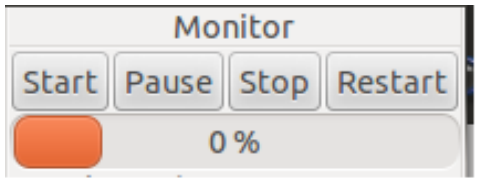
Générateur de topologie

- Charge et compile le générateur utilisateur



Contrôle de simulation

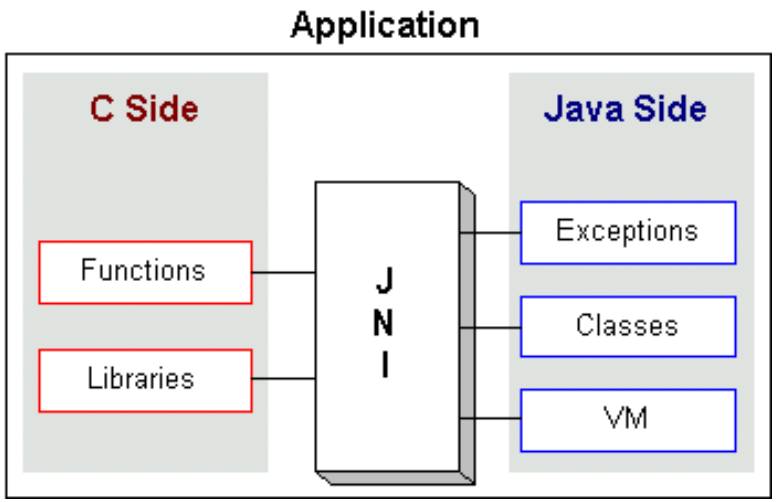
- Contrôle en temps réel
- Contrôle des simulations via JNI
- Retour utilisateur de l'avancement



Java Native Interface

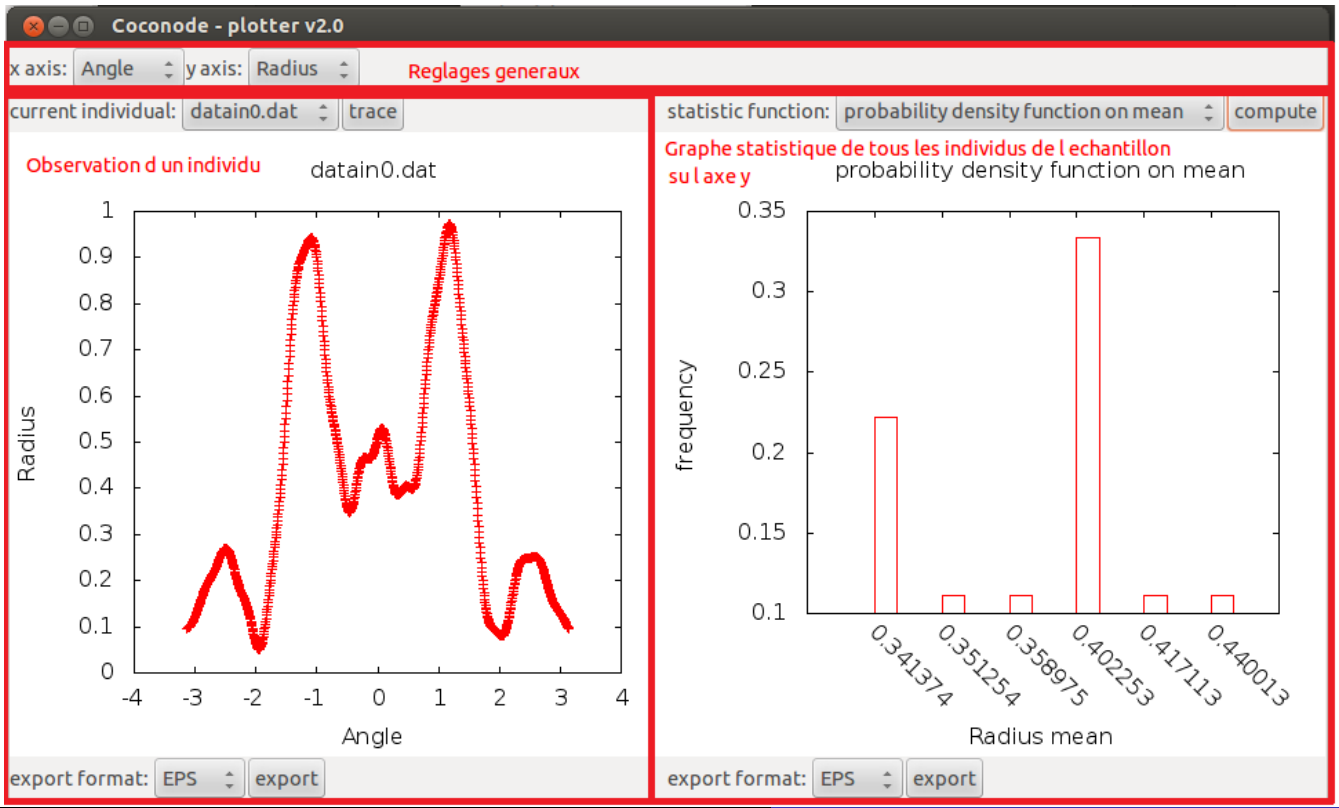
- Bibliothèque d'interfaçage du langage
- Java vers C
- Création une JVM
- Inclusion les sources de Cooja (.jar)
- Contrôle de Cooja

Cela donne accès aux classes de Cooja.



Afficheur de résultat/statistiques

- Choix des axes parsés dans le premier fichier de l'échantillon
- Liste des individus parsés dynamiquement dans le dossier d'entrée

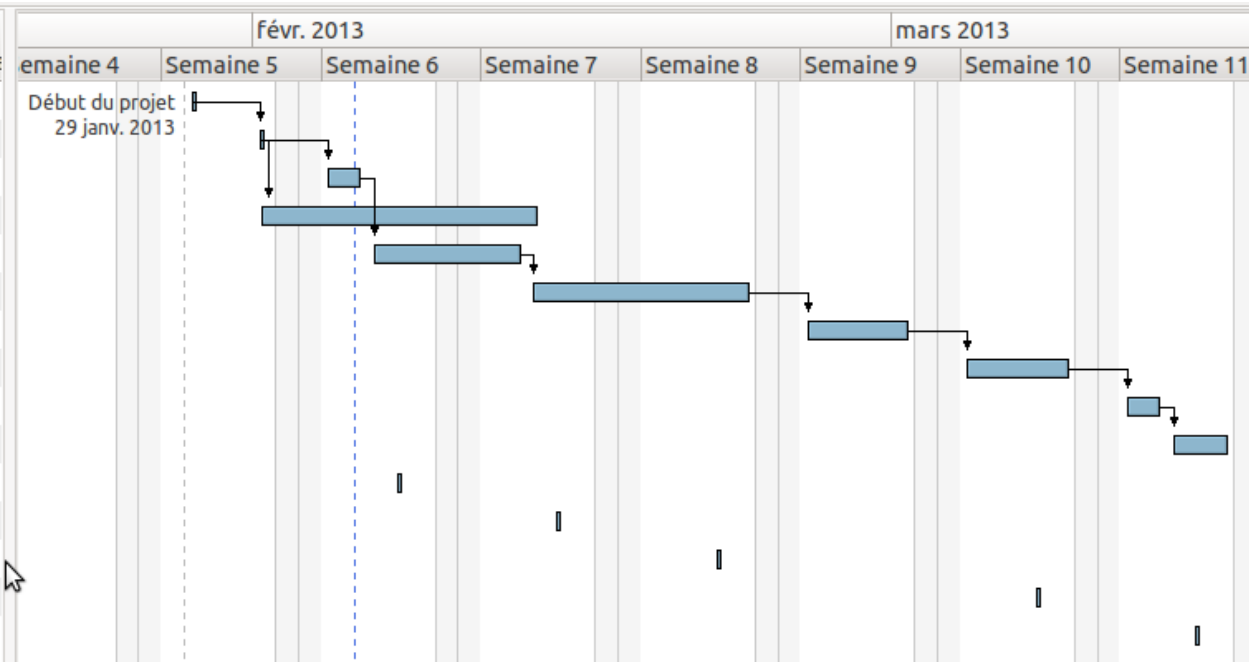


Outils et méthode utilisés

- Méthode Scrum
- Trello
- GIT, SVN
- Gantt (planning prévsionnel)

Planning prévisionnel Gantt

TPÉ	Nom	Trava
1	Prise de connaissance du projet	1j
2	Présentation de l'environnement	1j
3	Cahier des charges	2j
4	Prise en main de l'environnement	8j
5	Définition de l'architecture	5j
6	[Dev] partie résultat	8j
7	[Dev] partie configuration	5j
8	[Dev] partie monitoring	5j
9	[Dev] représentation de la topologie	2j
10	[Dev] Scheduler	3j
11	Réunion 1	2h
12	Réunion 2	2h
13	Réunion 3	2h
14	Réunion 4	2h
15	Réunion 5	2h



Pendant le Projet, par parties

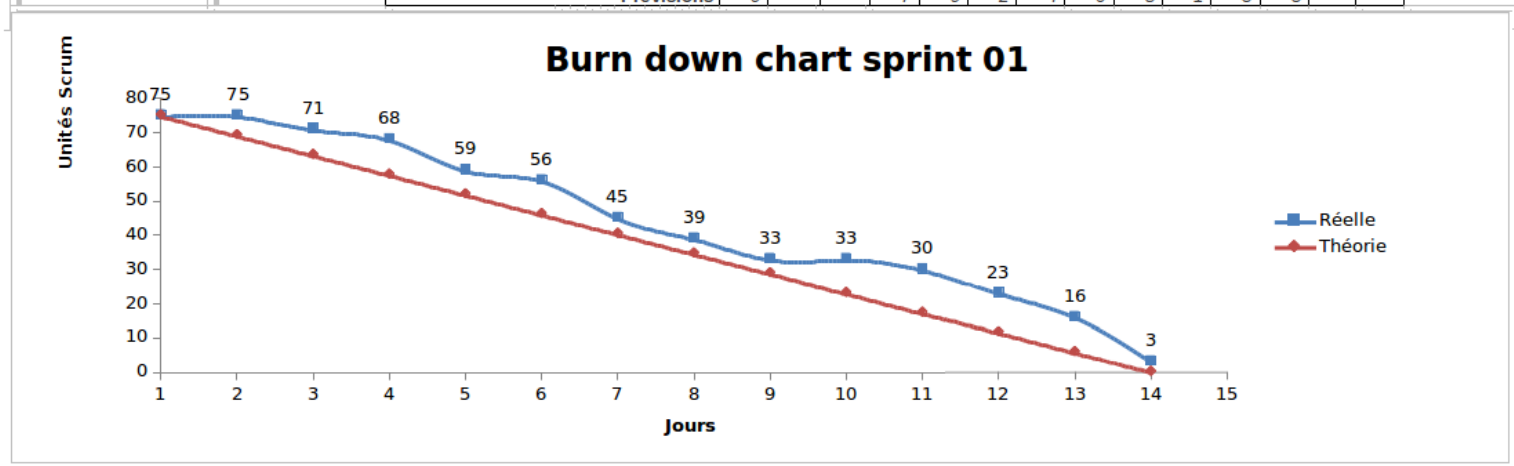
Responsabilité	Minh Quan Ho	Florian Lévêque	Noé-Jean Caramelli
Configuration	✔		
Scheduler	✔		
Générateur de topologie		✔	
Contrôle de la simulation		✔	
Afficheur statistique			✔
Tests Cooja			✔

Burn down chart du mardi 29 janvier au vendredi 15 février

Sprint 1 Coconode	Date de début	mardi 29 janvier 2013
	Date de fin	vendredi 15 février 2013
	Scrum Effort	75

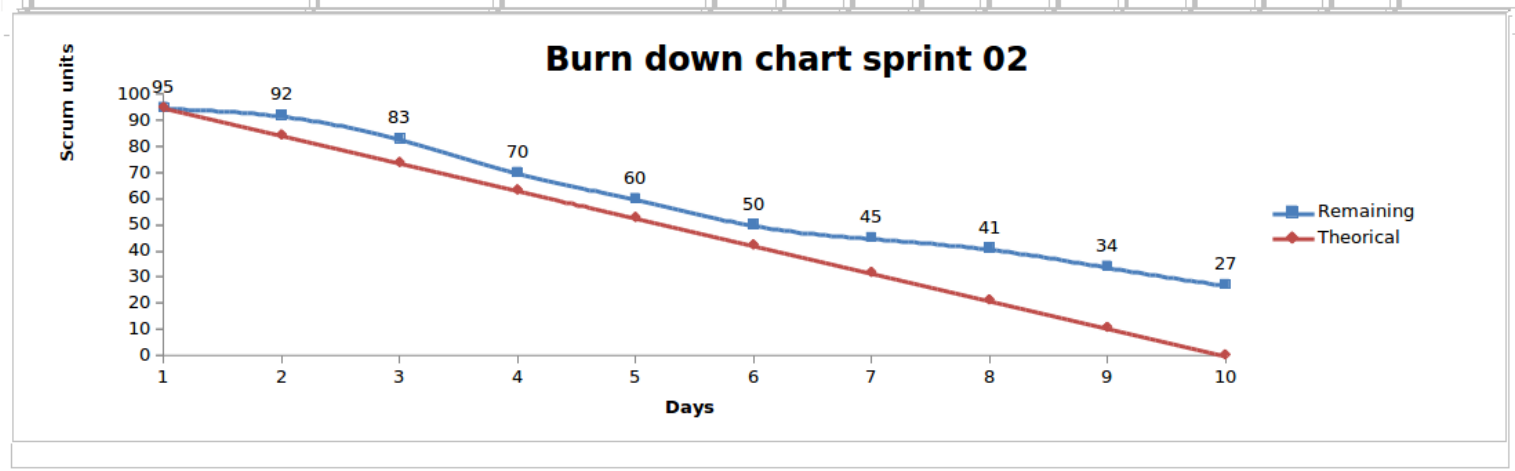
Unités Scrum restantes
3

Nom	Valeurs/Importance	Estimation initiale	mar	mer	jeu	ven	lun	mar	mer	jeu	ven	lun	mar	mer	jeu	ven
Cahier des charges	13	15	15	15	15	15	15	14	5	2	0	0	0	0	0	0
Architecture	13	25	25	25	25	25	22	22	22	22	20	15	13	10	10	3
Prototype de l'IHM	8	25	25	25	25	25	22	20	18	15	13	18	17	13	6	0
Prise en main de l'outil	10	10	10	10	6	3	0	0	0	0	0	0	0	0	0	0
		75	*75*	*75*	*71*	*68*	*59*	*56*	*45*	*39*	*33*	*33*	*30*	*23*	*16*	*3*
		Prévisions	*75,0*	*69,2*	*63,5*	*57,7*	*51,9*	*46,2*	*40,4*	*34,6*	*28,8*	*23,1*	*17,3*	*11,5*	*5,8*	*0,0*



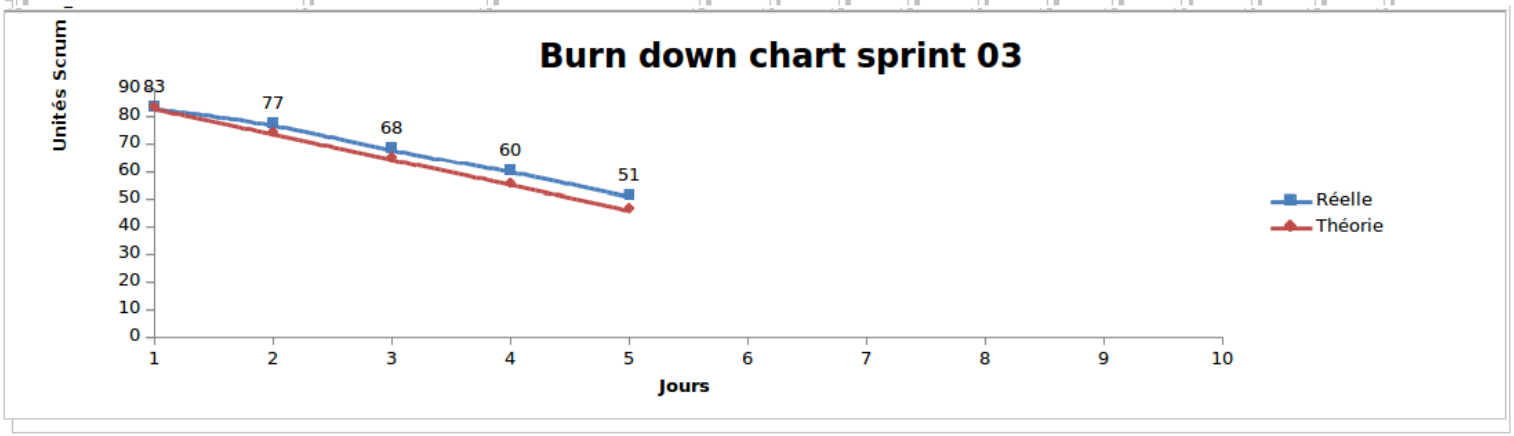
Burn down chart du lundi 18 février au vendredi 8 mars

Sprint 2	Date de début	lundi 18 février 2013										
	Date de fin	vendredi 8 mars 2013										
	Scrum Effort	95										Unités Scrum restantes
												27
Nom	Valeurs/Importance	Estimation initiale	lun	mar	mer	jeu	ven	lun	mar	mer	jeu	ven
Générateur de topologie	15	15	15	15	13	10	8	6	5	5	2	0
Configuration des simulations	30	10	10	7	3	0	0	0	0	0	0	0
Importer/Exporter données	25	20	20	20	20	20	15	12	10	7	5	2
Contrôle d'une simulation	35	25	25	25	25	20	20	17	17	17	17	17
Observation des résultats	35	25	25	25	22	20	17	15	13	12	10	8
		95	*95*	*92*	*83*	*70*	*60*	*50*	*45*	*41*	*34*	*27*
		Prévisions	*95,0*	*84,4*	*73,9*	*63,3*	*52,8*	*42,2*	*31,7*	*21,1*	*10,6*	*0,0*



Burn down chart du lundi 11 mars au vendredi 22 mars

Sprint 3	Date de début	lundi 11 mars 2013											
	Date de fin	vendredi 22 mars 2013											Unités Scrum restantes
	Scrum Effort	*83*											51
Nom	**Valeurs/Importance**	**Estimation initiale**	**lun**	**mar**	**mer**	**jeu**	**ven**	**lun**	**mar**	**mer**	**jeu**	**ven**	
Scheduler	**20**	**15**	15	15	15	15	15						
Monitoring	**5**	**3**	3	3	3	3	3						
Interruption d'une simulation	**20**	**10**	10	10	10	10	10						
Observation des résultats	**35**	**8**	8	8	7	7	5						
Contrôle d'une simulation	**35**	**17**	17	17	13	10	10						
Validation du logiciel	**30**	**15**	15	12	10	8	6						
Portabilité de l'outil	**20**	**15**	15	12	10	7	2						
		83	*83*	*77*	*68*	*60*	*51*						
		Prévision	*83,0*	*73,8*	*64,6*	*55,3*	*46,1*						



Points à finir

- doc doxygen
- doc utilisateur
- nettoyage du code
- tests de validation
- corrections et ajouts si besoin

Conclusion

Connaissances techniques acquises

- Techniques avancées de C
- GTK
- Gnuplot
- Bibliothèque JNI, glib, glib-json
- GIT
- Apprentissage Cooja, Contiki, WSN

Connaissances managériales acquises

- Mise en pratique de scrum
- Management d'une petite équipe
- Gestion de projet et support utilisateur
- Licences de publication
- Outils de communication (flyers, wiki)

Merci pour votre attention.

Passons à la démonstration.
Avez vous des questions ?