

COMPASS

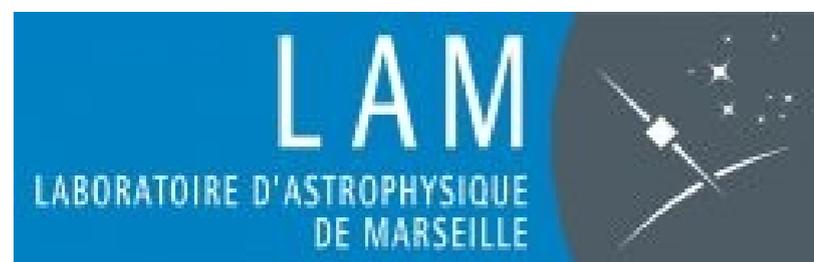
une plate-forme hétérogène haute performance pour le développement de l'optique adaptative de l'E-ELT



Laboratoire d'Études Spatiales et d'Instrumentation en Astrophysique



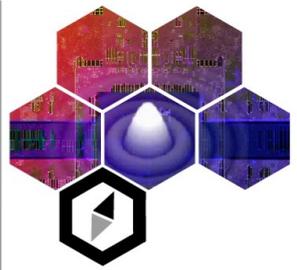
Galaxies Étoiles Physique et Instrumentation



MAISON DE LA SIMULATION

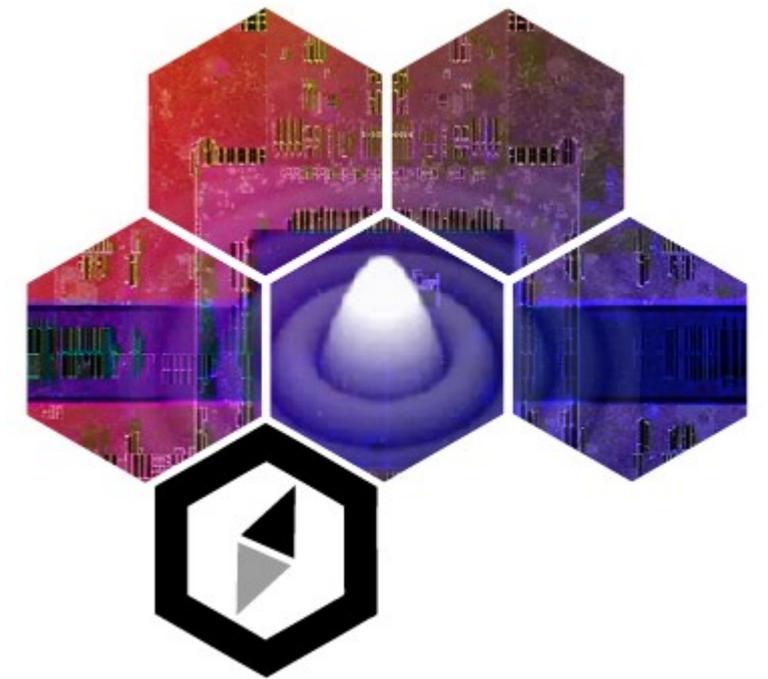


Ordre du jour



- 14h00 : Les messages de l'ANR (Cyril Demange)
- 14h45 : Présentation générale du projet (Damien Gratadour)
- 15h25 : Contribution du LESIA (Arnaud Sevin)
- 15h45 : Contribution du GEPI (Hector Flores)
- 16h05 : break
- 16h15 : Contribution de la MDLS (Pierre Kestener)
- 16h35 : Contribution de l'IPAG (Jean-Luc Beuzit)
- 16h50 : Contribution du LAM (Brice Leroux)
- 17h05 : Contribution de l'ONERA (Cyril Petit)
- 17h15 : Table ronde
- 18h00 : Fin

Kick-off COMPASS : Présentation générale du projet



Laboratoire d'Études Spatiales et d'Instrumentation en Astrophysique



Galaxies Étoiles Physique et Instrumentation

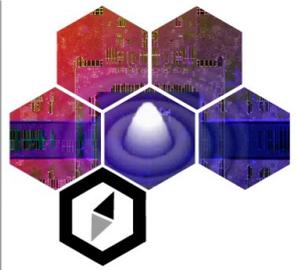


MAISON DE LA SIMULATION



Damien Gratadour
LESIA, Observatoire de Paris

Sommaire



- Motivations

- L'optique adaptative à l'ère de l'E-ELT
- Accélération matérielle et calcul haute performance

- Les objectifs scientifiques

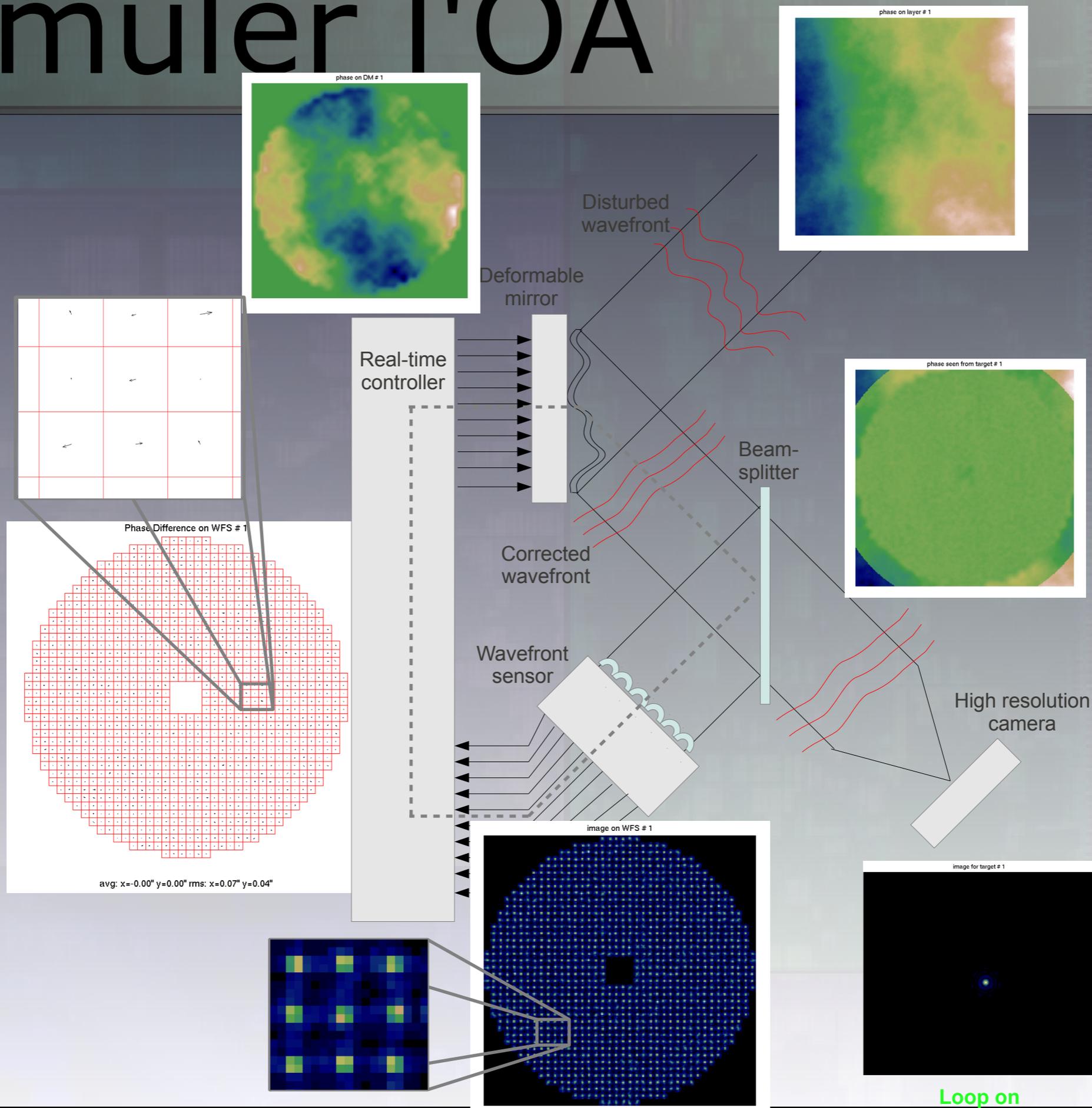
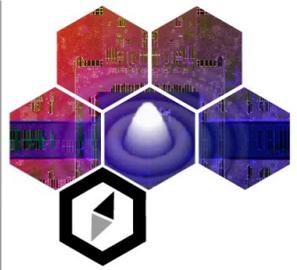
- Une plate-forme de développement unifiée et performante pour l'optique adaptative
- Une aide à la décision pour le design des futurs instruments de l'E-ELT

- Structure du projet et plan de travail

- Synergies entre simulations numériques et contrôle temps-réel
- Structure du projet
- Phases du projet

- Aspects organisationnels

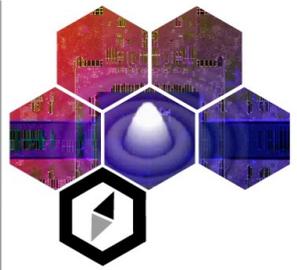
Simuler l'OA



Loop on

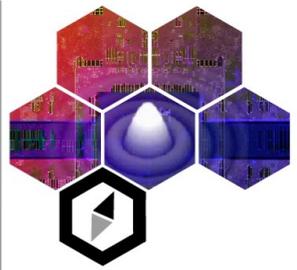
Loop off

État de l'art



- ◉ Outils de simulation « fait maison » utilisés dans différentes équipes
 - YAO (F. Rigaut) : code séquentiel pour CPU (Yorick / C) très utilisé. Modèle validé sur différents instruments (ALTAIR, MAD, GeMS). *Bundle* très complet (scripts + GUI)
 - Codes de simulation à l'ONERA, code séquentiel (IDL) validé sur différents instruments (NAOS, SPHERE), format ensemble de scripts
 - MAOS, code de simulation de la communauté US, initialement orienté CPU / multi-threads (migration Matlab vers C++), format librairie + scripts
- ◉ L'E-ELT : le passage à grande échelle
 - Projet européen de télescope de ~40m de diamètre
 - Finalisation du financement, démarrage prévu en 2013
 - Télescope entièrement adaptatif : OA au cœur de l'étude de concept
- ◉ Nécessité de fédérer les efforts de la communauté PHASE
 - Mener la recherche sur l'OA à cette échelle (nouvelles problématiques)
 - Designer et construire les instruments de l'E-ELT (retour scientifique)

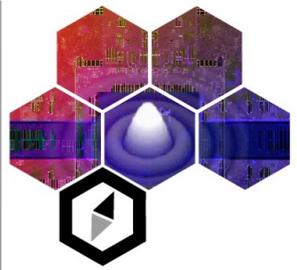
COMPASS : motivations



- Le diamètre de l'E-ELT fait de ces simulations un problème de grande échelle :
 - Pupille simulée: $\sim 2k \times 2k$ pixels (soit des écrans de phase de $20k \times 20k$ minimum)
 - Taille du support pour les TF rapides (formation d'images) : $\sim 4k \times 4k$
 - $\sim 5k$ sous-pupilles (ASO) and $\sim 5k$ commandes (MD) => plusieurs 100 GFLOPs nécessaires pour du MVM dans un schéma simpliste de contrôle temps-réel
 - Grand nombre d'itération pour atteindre la convergence (10-100k)
- Concepts avancés d'OA pour l'E-ELT (LGS MCAO, LTAO)
 - Plusieurs ASO et plusieurs MD, utilisation de guide laser
- Changement de paradigme :
 - Des modèles "fait maison" à des modèles unifiés à haute performance
 - Les simulations d'OA doivent entrer dans l'ère HPC

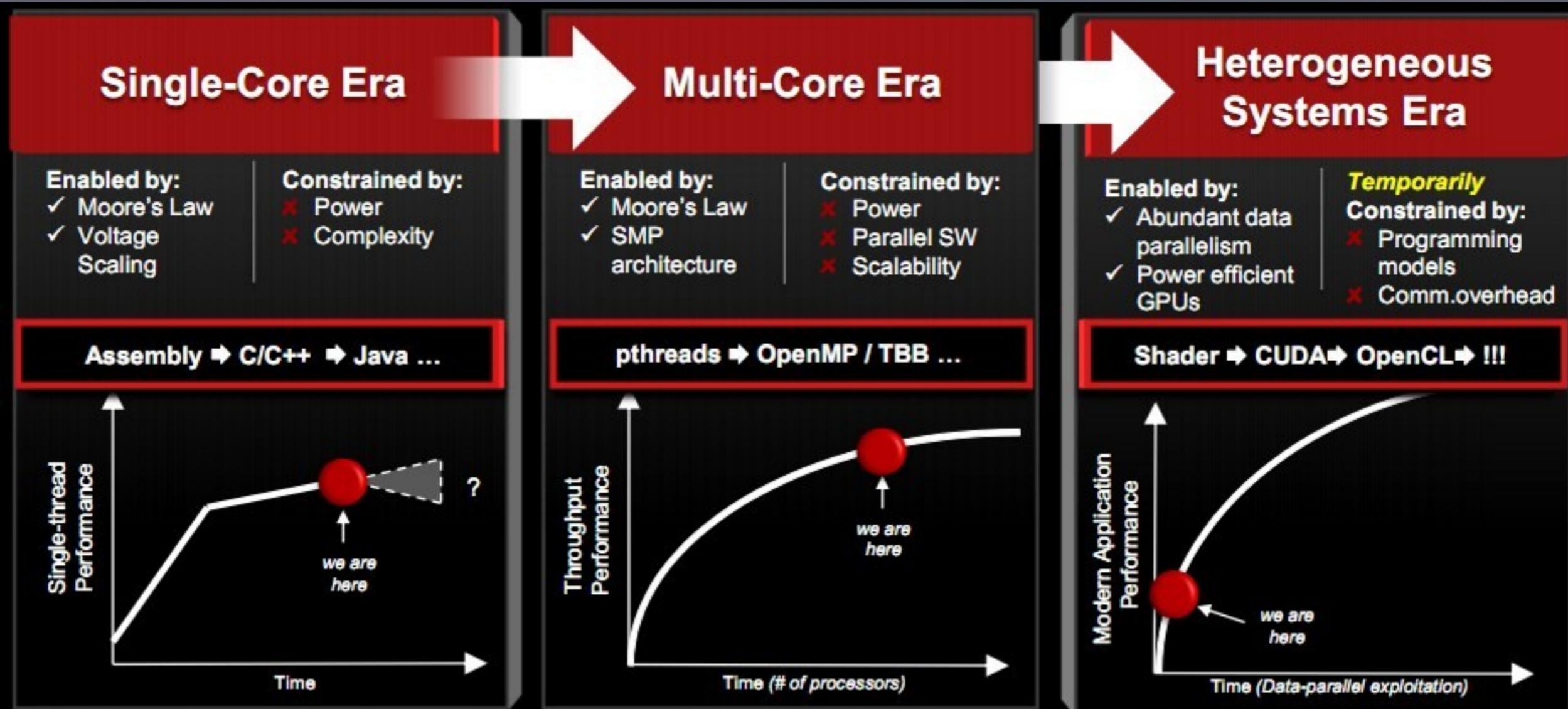
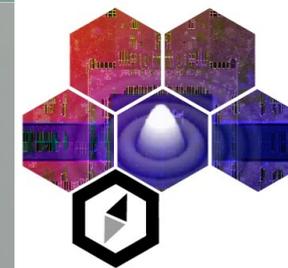


Accélération matérielle



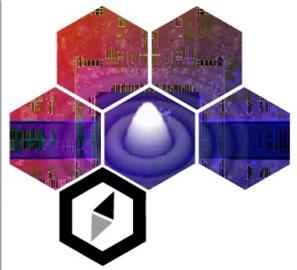
- ◉ Tendances actuelles dans l'industrie du HPC
 - La vitesse des processeurs n'augmente plus depuis le début du siècle (la fréquence d'horloge ne peut augmenter : chaleur, puissance, fuite de courant)
 - La concurrence est le nouveau paradigme : *hyper-threading, multi-core, many-core*
 - La prochaine révolution est avant tout logicielle : introduire de la concurrence
- ◉ Architecture hétérogène
 - Processeur multi-core généraliste (CPU) + coprocesseur spécialisé (DSP, GPU, MIC) ou FPGA + interconnexion réseau haute vitesse / faible latence
 - Programmation intrinsèquement complexe (différent hardware = modèle de programmation différent)
- ◉ *Many-core* : plusieurs approches
 - Utilisé des processeurs vectoriels réduits (Intel MIC, imminent)
 - Utiliser une grande quantité de petits processeurs scalaires dans un modèle SIMD (NVIDIA GPU, disponible)
 - Nécessite un parallélisme fin (+ vectorisation pour l'Intel MIC): "pas gratuit" !
 - Coprocesseurs: liés à travers le bus PCIe (bande passante + latence + modèle de mémoire)

Accélération matérielle



From "The death of CPU scaling" : <http://www.extremtech.com>

GPU



◉ NVIDIA Compute Unified Device Architecture (CUDA)

- 2000: premières cartes graphiques programmables = processeur à architecture unifiée, avec des cœurs scalaires (NVIDIA). Le terme GPU apparaît.
- 2003: l'idée d'un GPU généraliste (GPGPU) apparaît (brookGPU API)
- 2007: NVIDIA publie le premier environnement CUDA pour le GPGPU
- 2009: publication des spécifications OpenCL (non limitées à NVIDIA / aux GPUs)

◉ Développements de l'architecture des GPU

- *Streaming Multiprocessors* (SM) = groupe de cœurs de calcul. GPU = groupe de SMs + *scheduler*
- Dans un SM : unités SIMD = groupe *threads* exécutables (*warps* en CUDA)
- Plusieurs niveaux de mémoire : faible latence au niveau des SM (*registers, shared memory*) / latence plus grande au niveau de la puce (*global memory*)

◉ Ratio performance par W et par € élevé

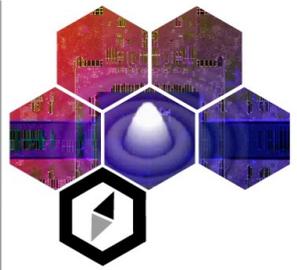
- Puissance de calcul > 1TFLOPs en simple précision pour quelques k€ et 100W
- Depuis nov. 2012, le serveur de calcul le plus puissant du monde: Titan (17 PetaFLOPs !) équipé de NVIDIA K20 (62 super-calculateurs dans le top500 utilisent GPUs)

YoGA



- ◉ 3 ans de développement au LESIA
 - Interfacer un langage de programmation haut niveau (Yorick) avec CUDA pour développer une simulation réaliste optimisée
 - ~X10 en performance comparé aux simulations « *single thread* »
 - Certaines parties du code montrent des facteurs plus importants (contrôle, supervision)
 - Interface complète et facile d'accès pour la réduction de données et *debugging*
- ◉ Boucle fermée !
 - Multiples écrans de phase à différentes altitudes / force / vitesse
 - multiples ASO type SH, guidage étoile naturelle ou laser
 - multiples MD à différentes altitudes
 - Simple algorithme de contrôle type « moindres carrés »
 - Premiers tests de simulation d'une SCAO sur l'E-ELT (MICADO)

YoGA



- Profils en ms sur une Tesla M2090 (système SCAO)

Telescope diam.	Turbo generation	Raytracing turbu	Raytracing DM	WFS	COG	Control	DM shape computation	Raytracing target
4m	0.107	0.008	0.008	0.138	0.013	0.019	0.137	0.008
8m	0.192	0.022	0.023	0.459	0.031	0.060	0.562	0.023
20m	0.550	0.135	0.136	3.07	0.079	0.363	3.22	0.137
30m	0.927	0.299	0.300	6.73	0.168	0.915	7.39	0.302
40m	1.44	0.526	0.525	11.9	0.320	2.263	13.62	0.527

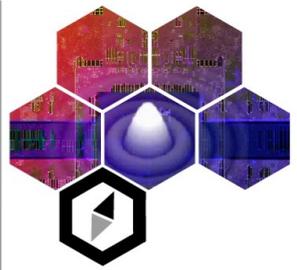
- Profil dominé par des tâches de simulation pures (modèles d'ASO et de MD)
- Performances des algorithmes centraux sur un seul GPU assurent du quasi temps-réel (échelle de l'ELT)
- Pas optimisé pour un GPU spécifique (*auto-tuning*)

YoGA



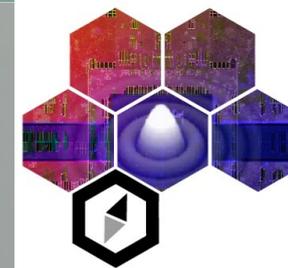
- ◉ 3 ans de développement au LESIA
 - Interfacer un langage de programmation haut niveau (Yorick) avec CUDA pour développer une simulation réaliste optimisée
 - ~X10 en performance comparé aux simulations « *single thread* »
 - Certaines parties du code montrent des facteurs plus importants (contrôle, supervision)
 - Interface complète et facile d'accès pour la réduction de données et *debugging*
- ◉ Sur cette base, les développements COMPASS
 - Prototype de modèle pour un ASO pyramide (à compléter)
 - Intégration de nouvelles lois de contrôle
 - Généralisation du mode multi-GPU (*peer-to-peer* + MPI)
 - Stabilisation, test, *debug*
 - Standardisation et maintenance à long terme

Environnement commun



- ◉ Avantages d'un environnement de développement commun
 - La puissance nécessaire est déjà disponible (en optimisant)
 - Pour la première fois une architecture peut être utilisée pour traiter les 2 aspects grâce à un environnement (logiciel + matériel) optimisé pour le GPGPU (CUDA / OpenCL)
 - Solutions plutôt économique basée sur du matériel disponible « sur étagère » avec un marché assez large et un environnement de développement gratuit (à comparer aux solutions FPGA / DSP)
 - Coût de développement réduit au minimum (à faire une fois seulement)
 - Le risque décroît alors que la robustesse et la capacité de mise à jour augmentent significativement
- ◉ Nécessité de s'attaquer aux différences fondamentales entre temps-réel et simulations (objectifs et contraintes)
 - Minimiser latence & *jitter*
 - Interagir de manière optimale avec le système (entrées / sorties)

Environnement commun



- ◉ Environnement unifié

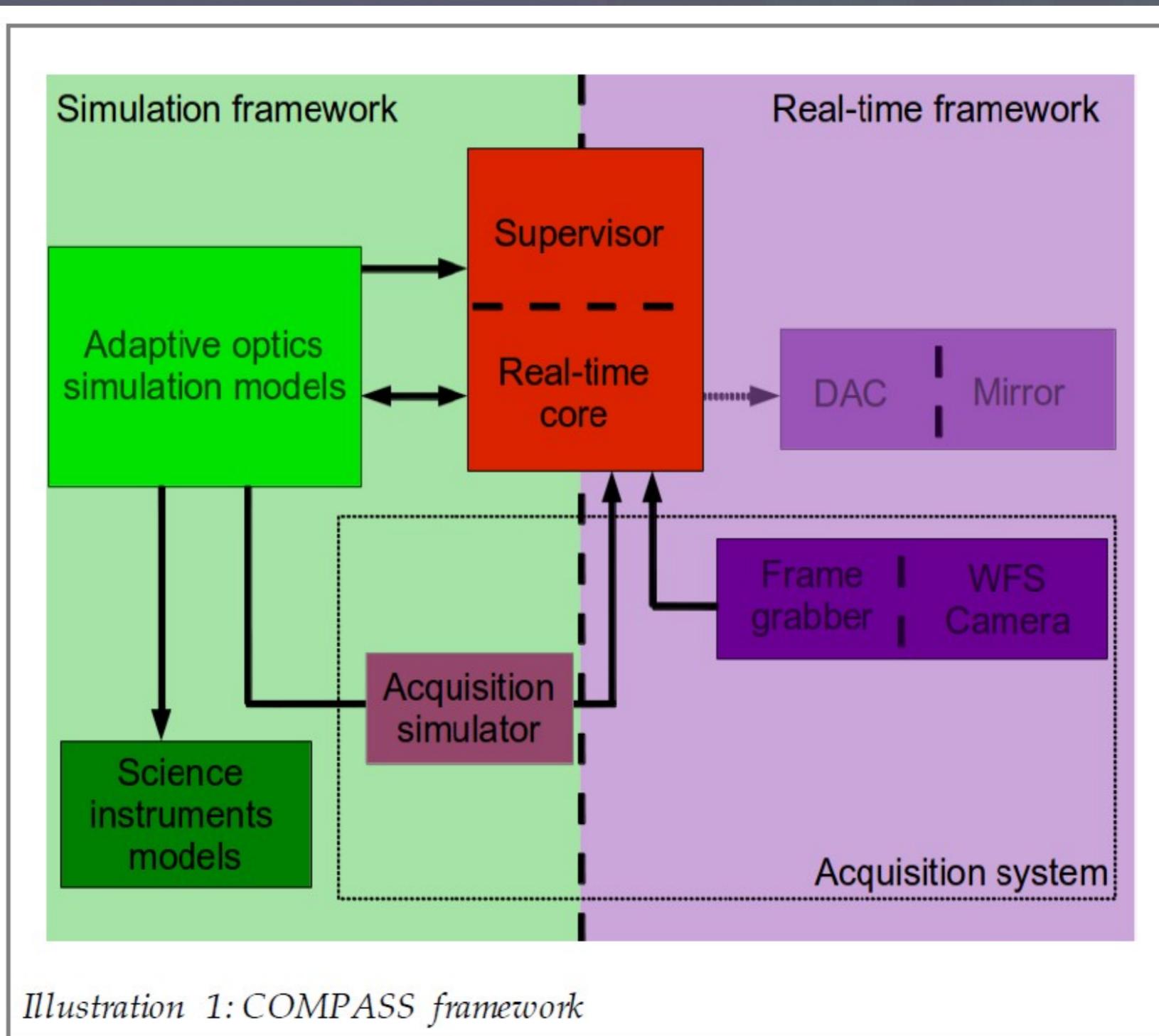
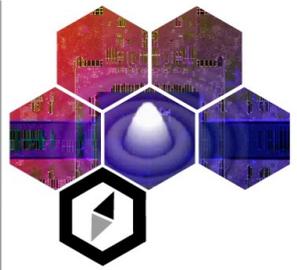


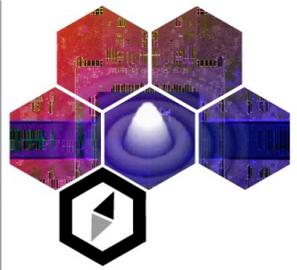
Illustration 1: COMPASS framework

Le projet COMPASS



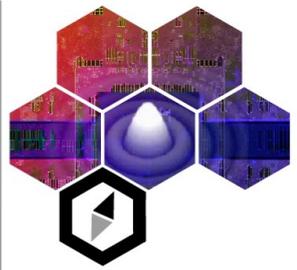
- COMputing Platform for Adaptive opticS Systems
- Un environnement unifié et optimisé sur une plate-forme hétérogène et évolutive
 - Fédérer les efforts au sein du partenariat PHASE (LESIA, GEPI, DOTA/ONERA, IPAG, LAM) pour développer et maintenir une plate-forme de développement numérique pour l'OA
 - Partenaire associé: *Maison de la simulation* un laboratoire regroupant 5 partenaires (CNRS, CEA et INRIA) pour la recherche en HPC
 - Collaboration multi-disciplinaire : OA + astrophysique + HPC
 - Produit final : une plate-forme haute performance basée sur une intégration totale du matériel et du logiciel et dédiée aux architectures massivement parallèles.
- Objectifs :
 - Plate-forme de développement logiciel : valider des composants clefs / tester de nouveaux concepts
 - Environnement de calcul efficace : mener des campagnes de simulations à large échelle pour le dimensionnement et l'étude de performance des futurs instruments
 - Prototypage des applications temps-réel : projet pilote pour le contrôle en OA basé sur l'accélération matérielle

Objectifs scientifiques

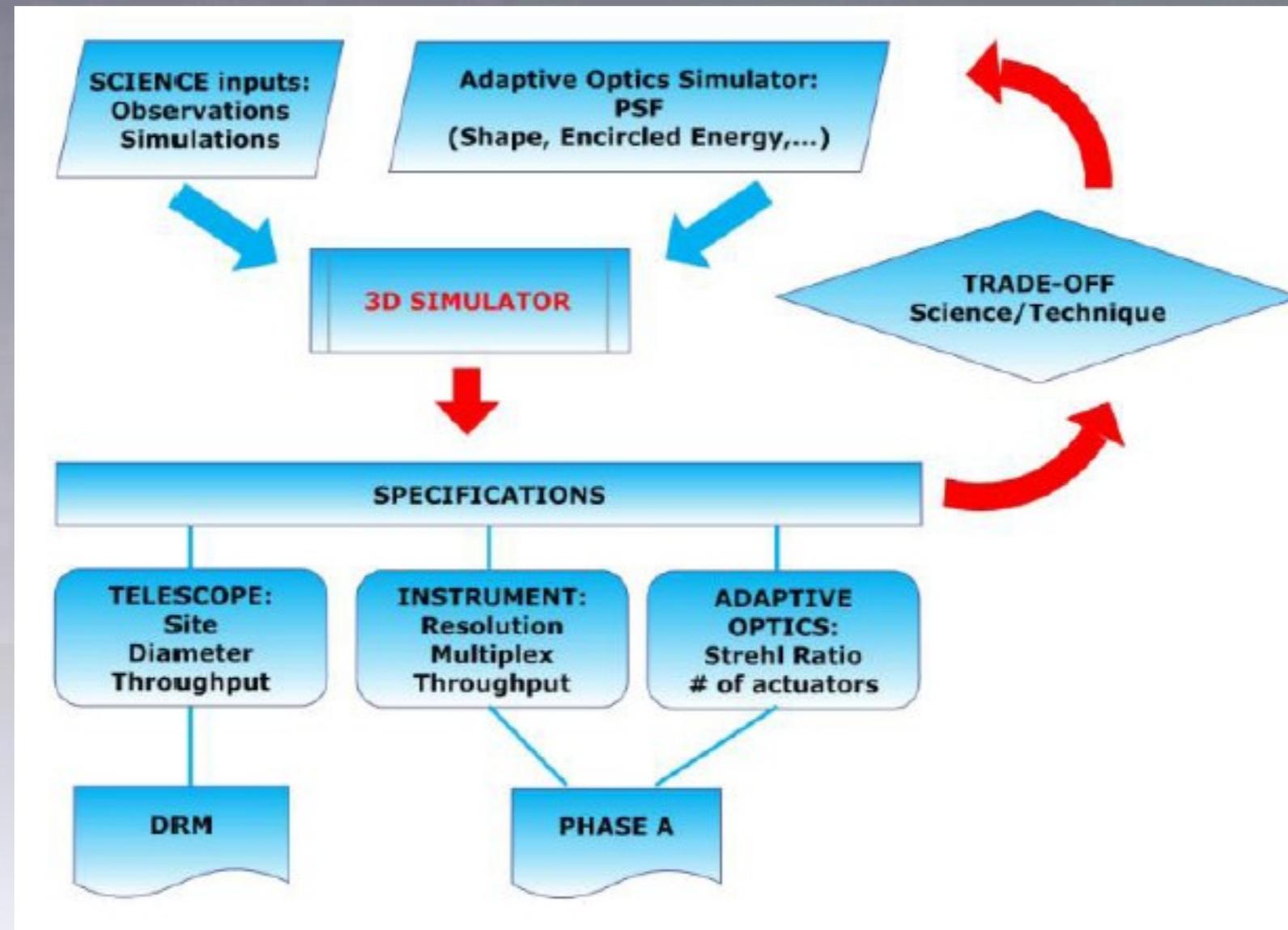


- Porter les algorithmes clefs
 - Optimisation des modèles mis en place : modèle de MD, modèle de pyramide
 - Porter et optimiser les algorithmes clefs pour le temps-réel : *Minimum variance*, LQG, *Learn & Apply*, méthodes itératives, méthodes Fourier
 - Porter et optimiser les stratégies de supervision: inversion de matrices creuses, gradients conjugués
- Développer une interface de transfert de données à faible latence
 - Le point clé du développement d'un RTC basé sur l'accélération matérielle (GPU, MIC)
 - Besoin de d'implanter un protocole d'adressage mémoire distant (RDMA) sur une interface série – PCIe (*framegrabber*) dédiée aux GPUs

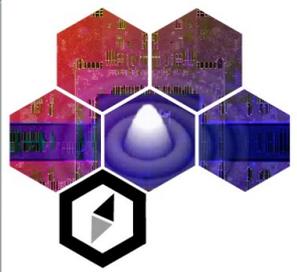
Objectifs scientifiques



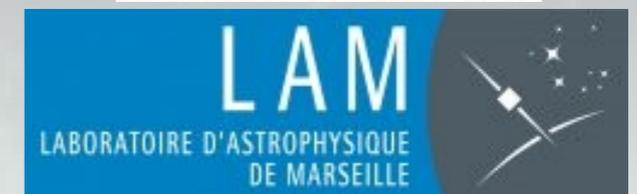
- Faire le lien avec l'instrumentation de l'E-ELT
- Plate-forme de simulation réaliste à l'échelle : de l'objet astrophysique à l'image corrigée par OA
- Aide à la décision pour le design des instruments : ELT-CAM (MICADO), ELT-IFU, ELT-MOS



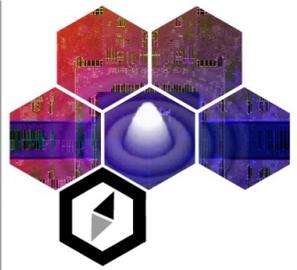
Structure du projet



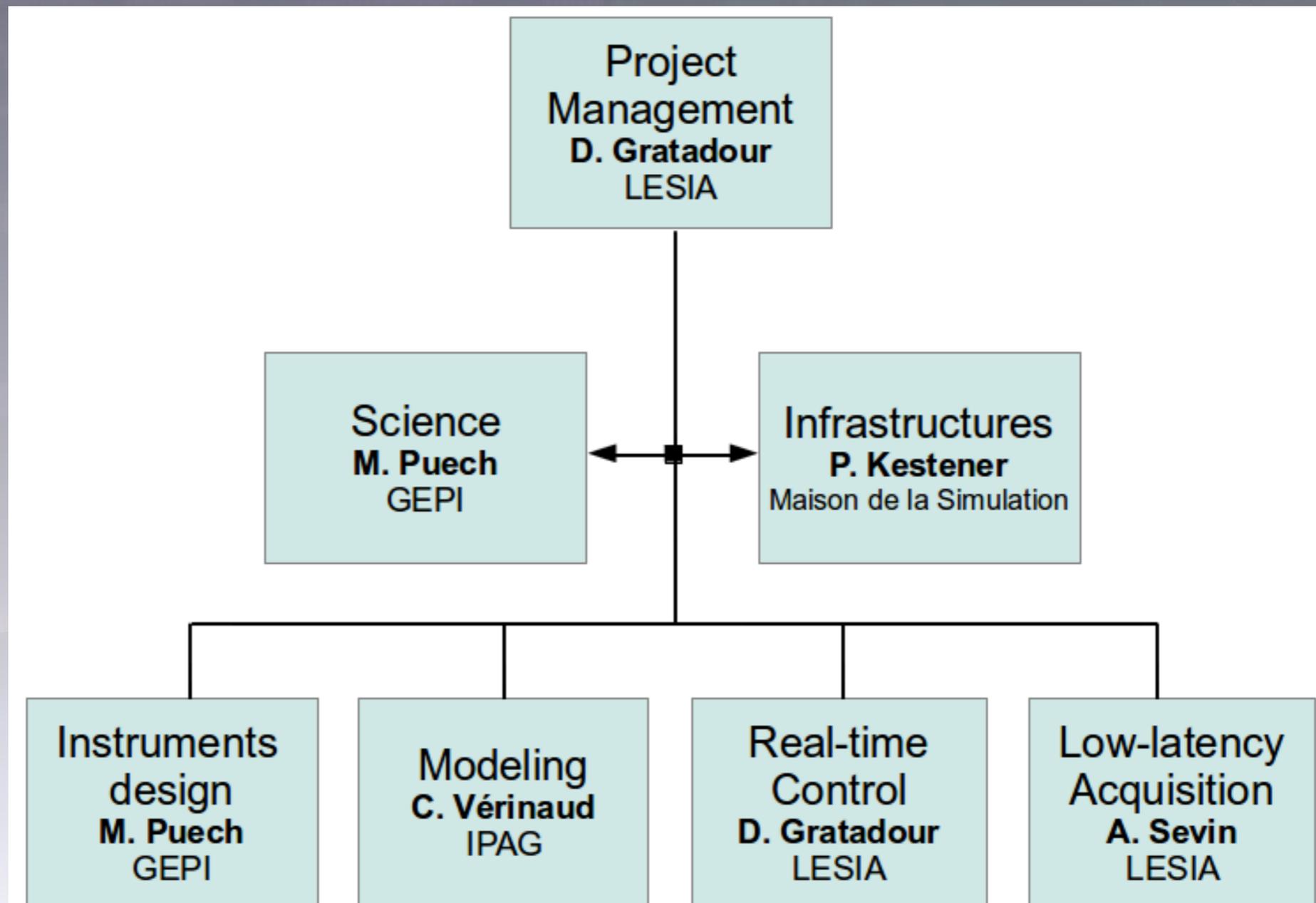
- 6 partenaires
- GIS Phase : LESIA, GEPI, IPAG, LAM, ONERA
- Maison de la simulation
- Management général : LESIA
- 2 groupes de travail
 - Science (GEPI)
 - Calcul (MDLS)
- 4 axes majeurs
 - Instruments (GEPI)
 - Modèles (IPAG)
 - Temps-réel (LESIA)
 - Interface faible latence (LESIA)



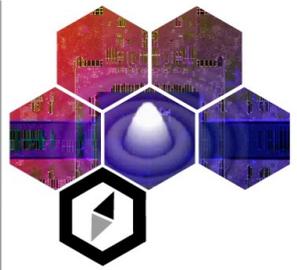
Structure du projet



- 2 pôles de supervision : science et infrastructures
- 4 axes majeures

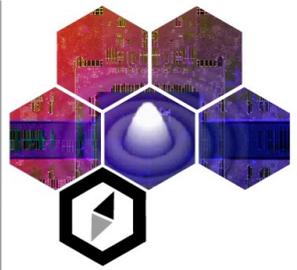


Intervenants clefs



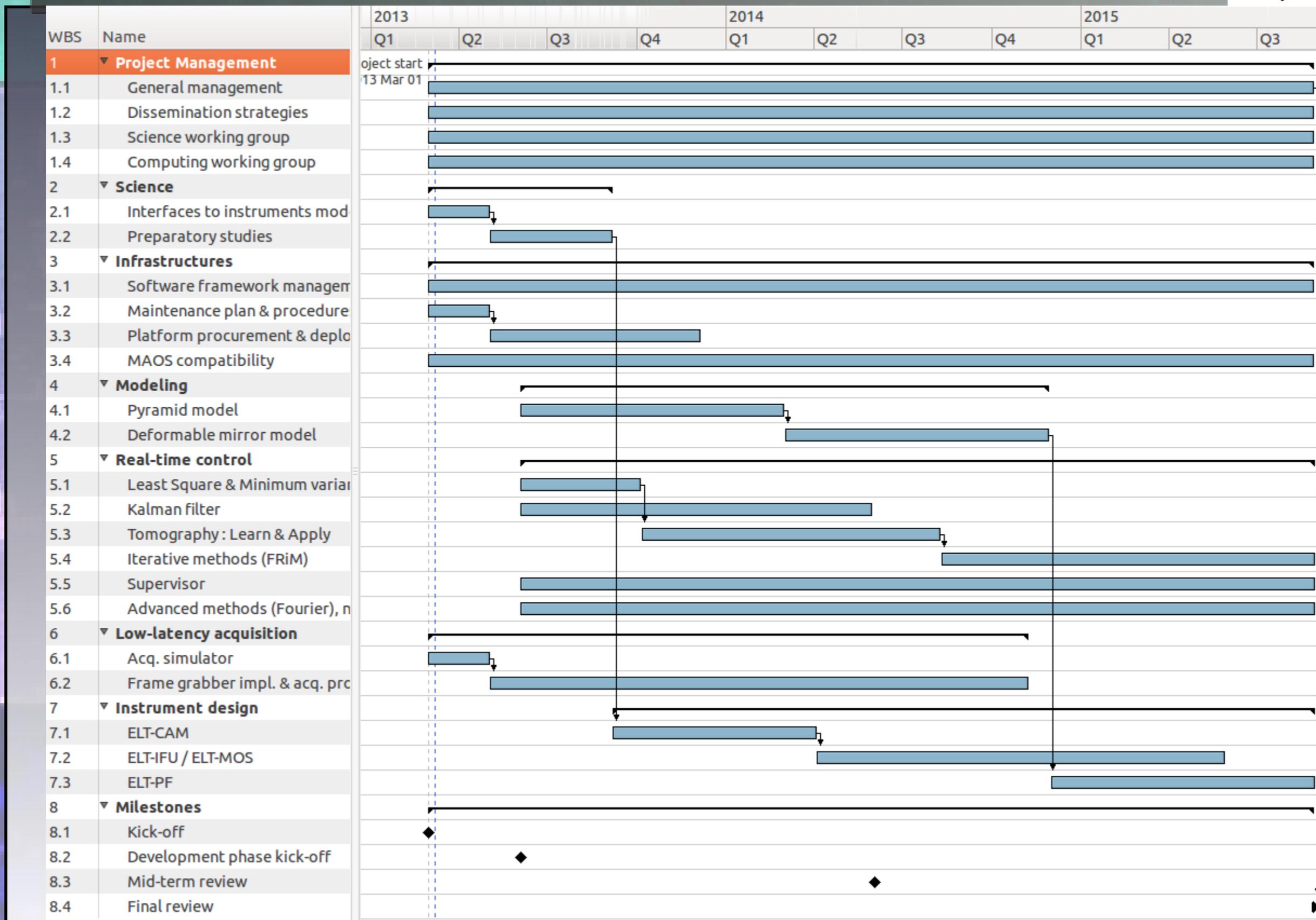
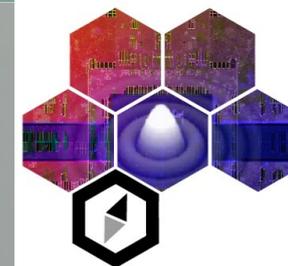
- LESIA :
 - A. Sevin (software / cuda), D. perret (électronique), J. Brulé (admin système / temps-réel)
- GEPI :
 - M. Puech, H. Flores, K. Disseau (modèles astro, strategie de dissemination)
- MDLS :
 - P. Kestener, M. Mancip (cuda, calcul haute performance)
- IPAG :
 - C. Vérinaud (OA, simulations, pyramide), J. L. Beuzit (modèles astro, OA, instruments)
- LAM :
 - B. Epinat (modèles astro), S. Rodionov (software), B. Leroux (OA)
- ONERA :
 - C. Petit, T. Fusco (OA, simulations)

Plan de travail

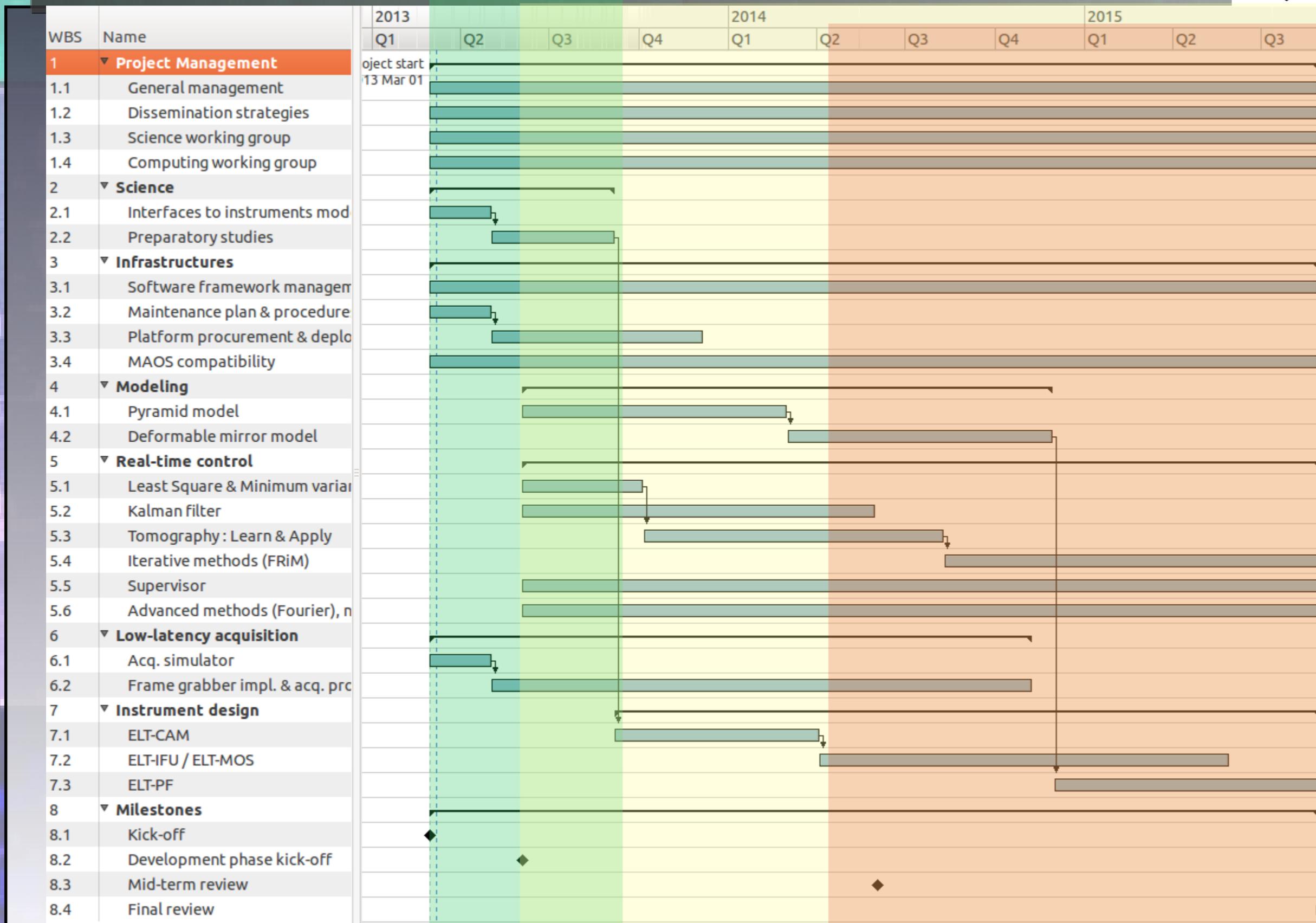
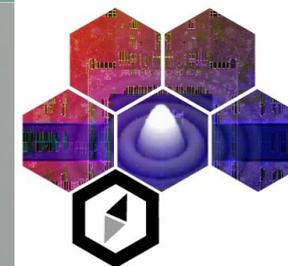


- 3 phases se recouvrant
 - Phase initiale de définition
 - Phase de développement
 - Phase d'exploitation
- 7 tâches principales
 - 1 : management du projet
 - 2 : science
 - 3 : infrastructures
 - 4 : modèles
 - 5 : algorithmes de commande
 - 6 : interface à faible latence
 - 7 : design de l'instrumentation pour l'E-ELT

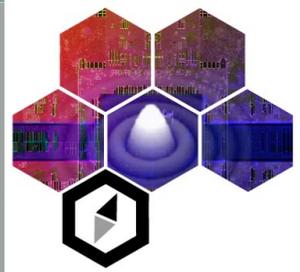
Plan de travail



Plan de travail

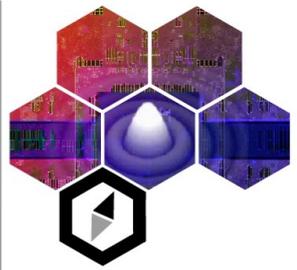


Plan de travail



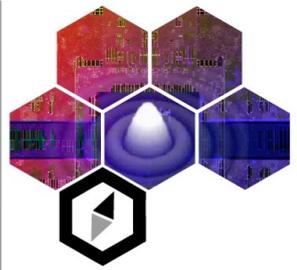
ID	Title & description	Date
M1	COMPASS kick-off	T0
M2	COMPASS development phase kick-off At this point, hardware & software platform are defined and maintenance plan is in place as well as the collaborative development framework and wiki	T0+3m
M3	Mid-term review: COMPASS framework validation At this point SWG and CWG will organize a mid-term reviews of both science (task 4 & 7) and computing (task 5 & 6) related activities and the overall platform should be deployed at each partner institute (<i>D3.3: COMPASS readiness review</i>)	T0+15m
M4	COMPASS final review	T0+30m

Organisation



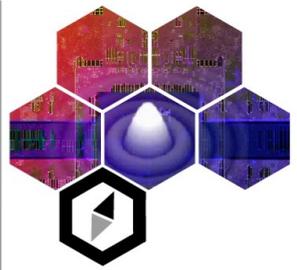
- Les 2 pôles de supervision (groupes de travail) interagissent avec les responsables des 4 axes majeurs
- Suivre l'avancement des tâches dans les axes concernés
- Organiser des réunions de progrès sur une base mensuelle
- Fournir les documents de synthèse de livrables
- Le management général coordonne l'activité des 2 pôles
 - Suivre l'avancement général du projet
 - Organiser l'interaction entre les 2 pôles (réunions de travail)
 - Synthèse de l'ensemble des livrables
- Répartition des tâches
 - Groupe de travail Science : axes instruments (tâche 7) et modèles (tâche 4)
 - Groupe de travail infrastructures : axes temps-réel (tâche 5) et interface faible latence (tâche 6)

Organisation



- ◉ Les 4 premiers mois (*deadline* juin 2013)
 - Réunion des groupes de travail pour lancer l'activité de définition
 - Science : définition des interfaces aux instruments et études préparatoire
 - Infrastructure : définition des infrastructures soft + hard, plan de maintenance
 - Fourniture d'un document de synthèse par groupe de travail
- ◉ Le kick-off de la phase de développement
 - ◉ Embauche des CDD + postdoc
 - ◉ Déploiement du matériel et démarrage des tâches de développement
 - ◉ Participation aux formations dédiées
- ◉ La revue à mi-parcours (juin 2014)
 - ◉ Premiers résultats d'exploitation
 - ◉ Documents de synthèse des 2 groupes de travail

Conclusions démarrage



- 14h00 : Les messages de l'ANR (Cyril Demange)
- 14h45 : Présentation générale du projet (Damien Gratadour)
- 15h25 : Contribution du LESIA (Arnaud Sevin)
- 15h45 : Contribution du GEPI (Hector Flores)
- 16h05 : break
- 16h15 : Contribution de la MDLS (Pierre Kestener)
- 16h35 : Contribution de l'IPAG (Jean-Luc Beuzit)
- 16h50 : Contribution du LAM (Brice Leroux)
- 17h05 : Contribution de l'ONERA (Cyril Petit)
- 17h15 : Table ronde
- 18h00 : Fin