

Hardware / software co-conception pour SKA

Shan Mignot



Laboratoire Lagrange, Observatoire de la Côte d'Azur, Université Côte d'Azur (UCA), Institut national des sciences de l'Univers
Scrum master pour l'équipe PlaNet au sein de l'effort SKAO

Introduction au Science Data Processor

- Partie intégrante des télescopes
 - 2 machines : en Australie pour Low, en Afrique du Sud pour Mid
 - dans la boucle de contrôle des queues d'observation
- Missions
 - reçoit les visibilité et les candidats pulsars produits par le Central Signal Processor
 - traite ces données pour permettre le stockage
 - génère des produits scientifiques
 - distribue ces produits aux Science Regional Centers et à l'archive SKAO
 - ne doit pas limiter la capacité des télescopes à observer

Activités du SDP

- 3 catégories de traitements
 - imagerie
 - recherche et mesure de pulsars
 - VLBI
- Une multiplicité d'autres activités
 - calibration: temps réel et maintenance des modèles
 - alertes
 - contrôle qualité
 - gestion de la ressource de calcul
 - archivage et distribution des données

Fonctionnement du SDP

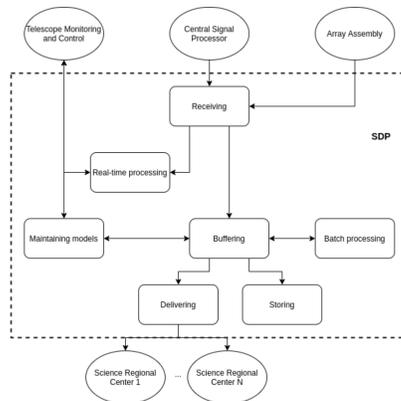
- Le SDP ne doit pas limiter la capacité des télescopes à observer
- Multiples processus : activités, fonctionnement en sous-réseaux, synthèse de faisceaux
- Estimations de charge (représentatif | maximum)
 - calcul : 10 Pflop | 125 Pflop
 - flux de données en entrée : 0,77 TB/s | 1 TB/s
 - accès mémoire interne : 4 TB/S | 10 TB/s
- Disponibilité
 - 99,9 % du temps
 - permettant une disponibilité des télescope > 95 % du temps

Objectifs pour le SDP

- Système évolutif
 - à court terme et pendant les 50 ans de la durée de vie de SKA
 - capable de passer à l'échelle : déploiement des antennes par phase puis extension des réseaux
 - favorisant l'évolution du logiciel et du matériel en séparant les expertises radio-astronomie et HPC
- Budgets (avec contingence)
 - disponible 128 M€
 - estimé : ~70 M€ pour les 2 machines & ~55 M€ pour le logiciel

Concept du SDP lors de la Critical Design Review

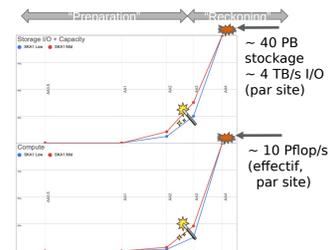
- Traitement massivement parallèle via une gestion indépendante des fréquences et du temps
 - passage à l'échelle via un accroissement du nombre de nœuds
 - impact à déterminer (par ex. pour la gestion des interférences)
- Architecture à 2 étages
 - algorithmes et workflow (radio-astronomie)
 - distribution, gestion de l'exécution et architecture matérielle (HPC)
 - impact à déterminer (par ex. optimisations algorithme/architecture)
- 2 modes opératoires : ingestion des données & temps réel, différé (batch)
 - lisser la charge sur plusieurs jours
 - impact à déterminer (par ex. planning des observations, cibles d'opportunité)
- Gestion de l'exécution
 - batch : par des approches de type dataflow
 - temps réel : via un moteur d'exécution spécifique (streaming)
 - pas de garantie sur la latence (plutôt une gestion en streaming des données avec un haut niveau de priorité)
 - impact à déterminer (par ex. latence min annoncée 15s)
- Uniformisation des serveurs et de leurs configurations
 - volonté d'homogénéiser les machines (redondance, coût, maintenance)
 - spécialisations à partir d'un même modèle de serveur
 - adaptations Low/Mid : I/O et nombre de serveurs
- Architecture
 - espace tampon (40 PB)
 - 4 types de serveurs dérivés d'un même modèle
 - ingestion et temps réel
 - traitements batch
 - services
 - stockage



Vue générale des flux de données au sein du SDP (l'interface directe avec les antennes n'est prévue que pour LOW)

Quelques inquiétudes

- Plan de gestion des risques issu de la CDR (2019) « SKA computing HW risk mitigation plan »
 - validité du modèle paramétrique utilisé pour le dimensionnement
 - hypothèse d'efficacité
 - extrapolation basée sur la loi Moore
- Évolution de la charge en fonction du nombre d'antennes
 - effet tardif et potentiellement catastrophique



Estimation de l'évolution des besoins en fonction du nombre d'antennes (d'après P. Wortman)

Consolider l'état des lieux

- Compléter l'évaluation CDR par du benchmark & profilage
 - Développer/mettre en place les outils
 - définir un jeu de cas de référence
- Étudier les impacts des choix de conception
- Évaluer la stratégie d'optimisation actuelle (par itérations)
 - par des mesures
 - via une stratégie d'extrapolation du logiciel et du matériel
 - avec une vision système

Activités de co-conception

- Optimisations itératives issues des benchmarks
- Modéliser les codes SKAO et simuler leur performances pour identifier des pistes d'optimisation logicielle et matérielle
- Prototypage ciblé pour évaluer les bénéfices et la difficulté d'adopter certaines pistes pour l'optimisation du système
- Développer une vision système du SDP
 - Contribuer à la spécification du SDP pour identifier/hiérarchiser les besoins et préparer l'appel d'offre
 - Développer un outil pour la comparaison de designs du SDP (performance, coût, risque etc. sur la base de la spécification du SDP et de directive de gestion de projet SKAO) pour évaluer la pertinence des pistes identifiées

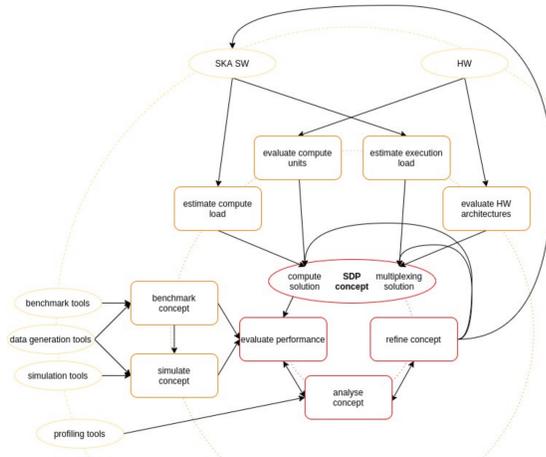


Schéma des interactions entre activités liées à la co-conception (le cercle extérieur représentant les données du problème)

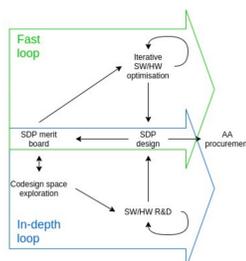


Illustration des deux processus itératifs de la co-conception

Quelques pistes

- Adopter une vision centrée sur les données
 - focus sur la réduction du volume au fil du workflow
 - réduction partielle des données
 - interrompt les itérations pour libérer des ressources
 - a priori et méthodes de régularisation vs. interprétation des données (par ex. émission diffuse vs modèle ponctuel)
 - publier des données partiellement réduites + résidus compressés
- Évaluer le bénéfice d'une spécialisation accrue des serveurs
 - ingestion et publication des données
 - prétraitements
 - gestion du tampon (~40 PB actuellement)
- Accélérateurs
 - déporter des tâches sur des unités spécialisées
 - hétérogénéité au sein des nœuds, au sein des unités (SoC)
- Contrôler la complexité pour mieux dimensionner le SDP
 - réduire la dispersion en complexité
 - potentiellement au prix d'une complexité minimale accrue
- Prédire la charge
 - pour évaluer l'intérêt de certaines pistes
 - pour fournir une estimation de disponibilité au télescope
 - intérêt pour la simulation SW&HW ou via l'instrumentation d'un benchmark
- Réduction du volume de données avant stockage sur le buffer (pré-processing) et gestion des besoins temps-réel