CONCEPTION DES LOGICIELS

TOME 3:
EXEMPLE DE DÉMARCHE DE CONCEPTION PRÉLIMINAIRE

Auteur: Bernard GIACOMONI

Version	Date	Objet
1.0	18/11/2021	Version initiale

Table des matières

I.INTRODUCTION :

Ce troisième tome de l'ouvrage CONCEPTION DES LOGICIELS présente une application de la démarche générale de conception préliminaire présentée par le tome II à un cas concret.

Ce cas a été choisi pour répondre aux impératifs suivants :

1. Permettre d'illustrer chacun des trois axes de conception exposés dans le tome II : la conception COMPORTEMENTALE, la conception SYSTÈME et la conception LOGIQUE ;
2. N'exiger pour sa compréhension aucune connaissance approfondie, ni dans le domaine concerné par le cas, ni dans des domaines connexes;
3. Ne pas impliquer un trop grand nombre de fonctionnalités ni des fonctionnalités trop complexes, afin que l'objectif premier de l'ouvrage, qui est l'application de démarches de conception ne se retrouve pas "dilué" dans un exposé trop volumineux;

En particulier, le cas a été choisi de manière à permettre l'illustration des points suivants de la démarche de conception préliminaire:

- Estimation des performances de la plate-forme d'accueil nécessaires pour satisfaire aux spécification de besoins de l'application ;
- Conception de cette plate-forme ou ajustement de la plate-forme existante pour satisfaire à ces performances ;
- Répartition des traitements sur les différentes unités de la plate-forme ;
- Estimation de la fiabilité et fiabilisation du fonctionnement.

Pour répondre à ces différents impératifs, le cas a été choisi dans le domaine de la conduite de processus. En effet, ce type d'application :

- Est le plus souvent réparti autour d'un réseau (pour des raisons de géographie des lieux, de fiabilité et de redondance) ;
- Exige des solutions multi processus et multi threads (pour permettre une réponse rapide et priorisée aux sollicitations externes);
- Présente un comportement assez complexe (tâches s'exécutant en parallèle, accès concurrents et aléatoires à des ressources communes) ;
- Intègre la plupart du temps des mécanismes de fiabilisation (redondances, répartition de charge, détection de dysfonctionnements, modes dégradés, etc) ;

En compensation, le cas choisi pourra présenter un aspect un peu artificiel car, par soucis

d’être compréhensibles par tous sans exiger de recherche ou d’études complémentaires et de diminuer la complexité algorithmique, les traitements exigés au niveau de la spécification des besoin sont très sommaires par rapport à ce qu’ils seraient dans une véritable application du domaine.

II.APPLICATION DE LA DÉMARCHE DE CONCEPTION À UN CAS CONCRET :

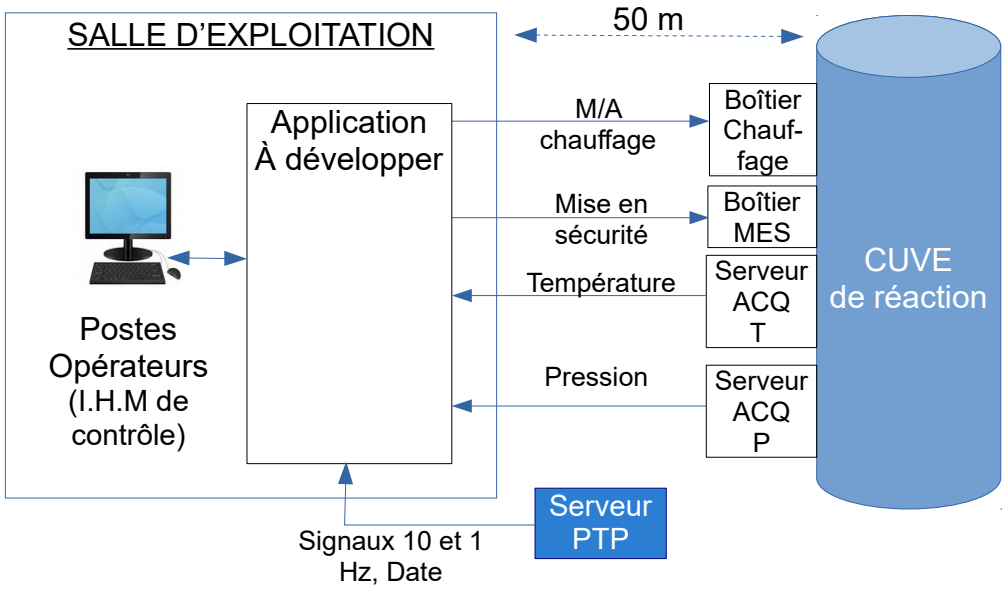
II.1.SPÉCIFICATION DU BESOIN:

II.1.1.DESCRIPTION GÉNÉRALE :

- L'application a pour mission de contrôler une réaction chimique se produisant dans une cuve (réacteur) ;
- Cette cuve est chauffée par un dispositif électrique dont l'alimentation peut être soit établie, soit coupée par une commande transmise par l'application ;
- La salle de contrôle abritant le système informatique de conduite du processus et les postes d'exploitation est située, pour des raisons de sécurité, à une distance de 50 m de la cuve;
- Un boîtier électronique intégré à la cuve commande le chauffage de la cuve. Il supporte une commande manuelle marche/arrêt et une connexion réseau permettant de recevoir cette commande par un message réseau UDP/IP ;
- Un autre boîtier électronique sécurisé, intégré à la cuve, permet le déclenchement d'un mécanisme de mise en sécurité (MES) de la cuve en cas de dysfonctionnement. Ce mécanisme, qui assure la dépressurisation et le refroidissement de la cuve, peut être commandée manuellement (à partir du boîtier) ou par un signal externe de type on/off que le boîtier peut acquérir sous la forme d'un message réseau UDP/IP.
- Le site dispose d'un serveur PTP (Precision Time Protocol) capable de fournir au système informatique une datation à la milliseconde ainsi que divers signaux de synchronisation (signaux 10 hz, 1 Hz, etc.) ;
- La pression et la température à l'intérieur de la cuve sont mesurées toutes les 100 ms rondes (10 hz) par des capteurs. Ces données sont transmises au logiciel de contrôle par l'intermédiaire de boîtiers "serveurs d'acquisition" délivrant ces mesures sous forme de messages réseau en mode UDP multicast. Ces données sont disponibles sur le réseau au plus tard 5 ms après chaque signal "10 Hz" du PTP.

DOC: Conception des logiciels. Tome III

Le schéma suivant résume les caractéristiques matérielles et logiques du système :



REMARQUES:

- Le contrôle de la réaction consiste à maintenir à l'intérieur de la cuve une température proche de la température de consigne saisie par les exploitants grâce à l'IHM du système) ;
- Les exploitants, informés en temps réel de l'évolution de la température et de la pression, peuvent lancer à partir d'une commande IHM la mise en sécurité de la cuve.

II.1.2.ÉNONCÉ DES EXIGENCES:

EXIGENCE N°1		
ÉNONCÉ	Le **PERSONNEL D'EXPLOITATION** doit pouvoir **CONTRÔLER EN TEMPS RÉEL** l'**ÉTAT COURANT DE LA CUVE** sur l'écran de son IHM.	
N° Critère	Domaine	Caractérisation
1.1	Données consommées	Température et pression issues de capteurs: - Fréquence d'acquisition = 10 Hz; - Format: entier positif ou nul sur 16 bits (fp = 1 hp pour la pression et 0,1 degrés pour la température).
1.2	Utilisateurs concernés	Personnel d'exploitation. - Localisation: salle d'exploitation de l'usine, située à 50 m de la cuve (distance de sécurité en cas d'explosion) ; - Nombre d'utilisateurs simultanés: 1.
1.3	Équipement d'affichage	Écrans graphiques.
1.4	Données à visualiser	- Nature : moyennes arithmétiques sur les 11 dernières mesures acquises de la température et de la pression ; - Forme: cadrans ronds (objet graphique) gradué en centaines d'hectopascals pour la pression et en degrés pour la température - Précision > 1 hp pour la pression, 0,1 degrés pour la température ; - Fréquence de rafraîchissement = 10 Hz; - Délais d'affichage de moins de 200 ms entre l'acquisition de la plus récente donnée participant la moyenne et l'affichage de cette moyenne; - Absence de mesure: affichage valeur 0.
1.5	Fiabilité, disponibilité.	Probabilité de panne sur une année d'exploitation : < 10^{-2}
1.6	Maintenabilité.	La remise en état doit pouvoir s'effectuer "à chaud".
1.7	Sécurisation des accès	Accès réservé aux personnels authentifiés par Id. et mot de passe.

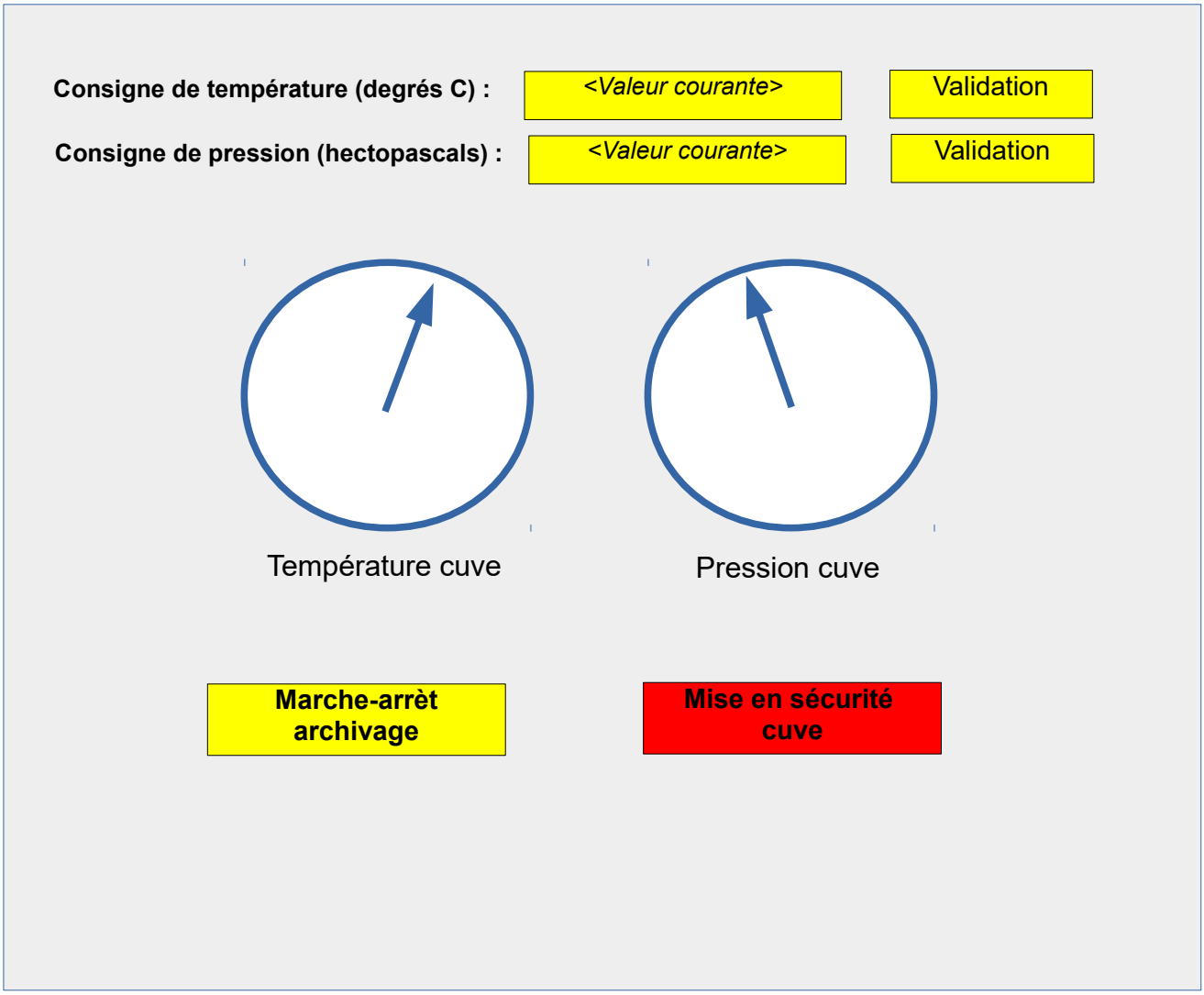
EXIGENCE N°2		
ÉNONCÉ	Le **PERSONNEL D'EXPLOITATION** doit pouvoir **ARCHIVER** sur un support permanent les **TEMPÉRATURES** et **PRESSIONS** acquises ainsi que leurs **DATES D'ACQUISITION**.	
N° Critère	Domaine	Caractérisation
2.1	Fréquence d'archivage	1Hz
2.2	Données à archiver	Archivage sous forme de texte. Un bloc d'archivage : - Date et heure d'archivage (Date, H/M/S/10^e) - Tableau des 10 dernières températures acquises+Dates d'acquisition (précision : 1/10^e de seconde). - Tableau des 10 dernières pressions acquises+Dates d'acquisition (précision : 1/10^e de seconde).
2.3	Commande marche/arrêt de l'archivage	Par un bouton graphique activable sur l'écran de contrôle de l'IHM.
2.4	Repérage des archives	Au démarrage de l'enregistrement, un nouveau fichier texte est crée. Sa structure est : <date de démarrage (date posix)>+".reg"
2.5	Fiabilité, disponibilité	Probabilité de panne sur une année d'exploitation : $>10^{-2}$
2.6	Maintenabilité	Le remplacement d'une unité d'archivage défaillante doit pouvoir être effectué "à chaud".

EXIGENCE N°3		
ÉNONCÉ	Le **PERSONNEL D'EXPLOITATION** doit pouvoir **DÉFINIR** et **CONTRÔLER** les **VALEURS DE CONSIGNE** de la température et de la pression de la cuve.	
N° Critère	Domaine	Caractérisation
3.1	Données de consigne	Température et pression de consigne de la cuve.
3.2	Méthode de saisie	Champs de saisie sur l'écran de l'IHM : - Format: Flottant positif ou nul: - Unités: pression = hecto pascal, température = degrés ;
3.3	Définition des valeurs de consigne	Au moyen de deux champs de saisie accessibles sur l'IHM d'exploitation. Saisie en degrés C et hectopascals ;
3.4	Contrôle des valeurs de consigne	Les deux champs de saisie affichent les valeurs courantes des consignes de température et pression, en degrés et hectopascals.

EXIGENCE N°4		
ÉNONCÉ	Le **PERSONNEL D'EXPLOITATION** doit pouvoir à tout moment **METTRE EN SÉCURITÉ** la cuve ;	
N° Critère	Domaine	Caractérisation
4.1	Moyen de contrôle	Bouton graphique activable sur l'écran de l'IHM déclenchant un message vers le boîtier de mise en sécurité.
4.2	Délais de réaction	< 100 ms (hors temps de réaction humain)

EXIGENCE N°5		
ÉNONCÉ	Le **SYSTÈME** doit **MAINTENIR AUTOMATIQUEMENT** la **TEMPÉRATURE DE LA CUVE** au voisinage de la **TEMPÉRATURE DE CONSIGNE**.	
N° Critère	Domaine	Caractérisation
5.1	Précision de la régulation (écart tolérable).	± 0,3°
5.2	Source des données de régulation	- Température de la cuve : représentée par la valeur moyenne calculée sur les 11 dernières températures acquises ; - Température de consigne : dernière valeur de consigne saisie par l'IHM.
5.3	Durée entre la détection d'un écart non tolérable par rapport à la consigne et la réception de la commande M/A chauffage de correction de cet écart par le boîtier de contrôle du chauffage.	< 50 ms
5.4	Fiabilité, disponibilité	- Probabilité de panne sur une année d'exploitation: $> 10^{-4}$; - Le dysfonctionnement d'une unité de stockage ne doit pas occasionner de pertes des mesures acquises pendant la remise en état.
5.5	Maintenabilité	Remise en état à chaud.

Schéma de principe de l'IHM :



III.CONCEPTION GÉNÉRALE :

III.1.VALIDATION-COMPLÉTION DE LA SPÉCIFICATION DU BESOIN :

Cette étape ne concernant pas les activités de conception, elle ne présente aucun intérêt dans le cadre du présent volume. Nous la considérerons donc comme acquise.

III.2.ÉTUDE COMPORTEMENTALE :

III.2.1.ÉTUDE DES ACTIVITÉS DE L'APPLICATION :

III.2.1.1.ANALYSE DES ACTIVITÉS INDUITES PAR CHAQUE EXIGENCE :

Nous allons déduire directement les activités de l'application des exigences exprimées par la S.T.B en utilisant les diagrammes d'activité d'U.M.L. ***:***

RAPPELS : Les différents nœuds figurant dans ces diagrammes sont :

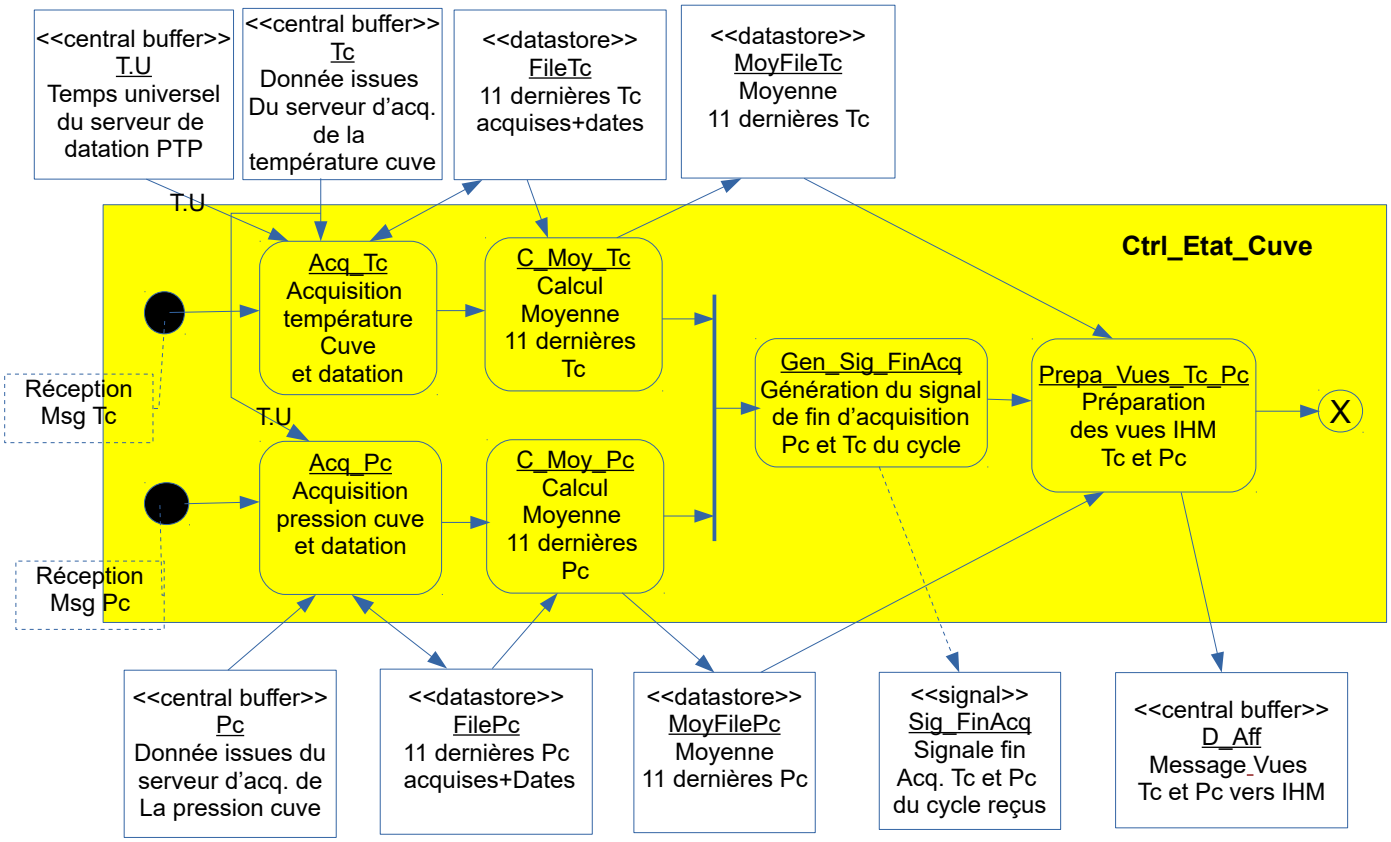
- Des NŒUDS D'ACTIONS (rectangles à coins arrondis) qui représentent des traitements que l'on peut considérer comme "atomiques";
- Des NŒUDS D'OBJETS (rectangles), qui représentent des flux de données ;
- Des NŒUDS DE CONTRÔLE, qui permettent de représenter la logique du chemin d'exécution (débuts et fins de flux, nœuds de décision ou de fusion, etc.).

REMARQUES :

- Les nœuds d'actions figurant dans ces diagrammes peuvent représenter des traitements beaucoup plus complexes que ceux qui sont préconisés dans le cadre de la notation U.M.L. Cependant, il s'agit de traitements qui, au niveau d'analyse où nous nous trouvons, peuvent être suffisamment caractérisés par une simple phrase (libellée à l'intérieur du pictogramme d'action). De ce fait, nous les assimilerons à des actions au sens d'U.M.L ;
- Les contenus des flux décrits par les nœuds d'objets, sont également spécifiés à l'intérieur de ces pictogrammes. Rappelons la signification des stéréotypes :
  - <<centralbuffer> définit un "tampon" (une file d'attente de messages) qui peut être accédée par plusieurs actions en entrée et en sortie ;
  - <<datastore>> définit une zone de données pouvant être partégée par plusieurs activités ;
  - <<signal>> définit un signal émis par une activité source et pouvant être transmis à plusieurs activités destinataires ;
- Ces entités sont décrites plus explicitement dans des tableaux de synthèse situés ci-après.

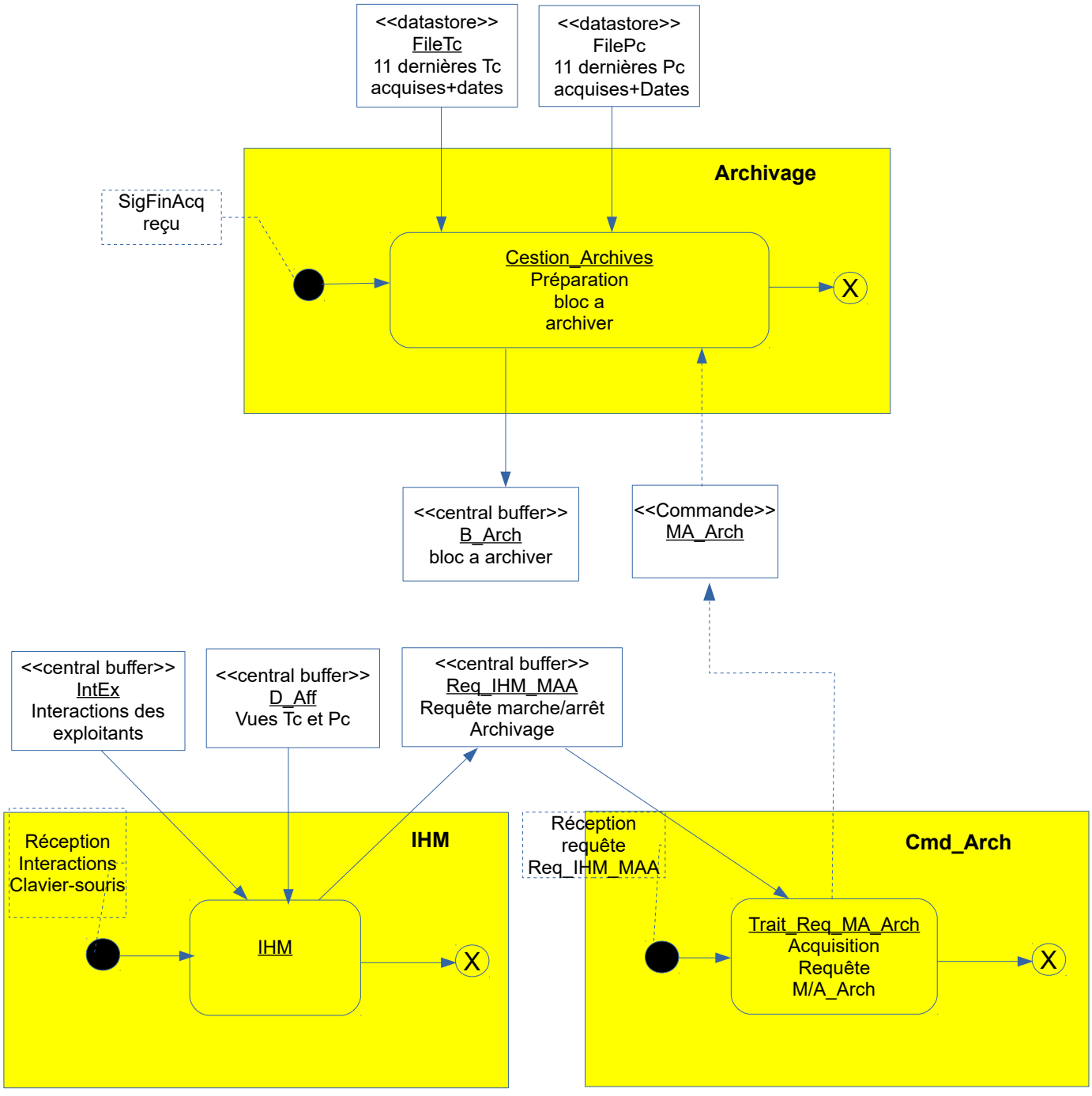
III.2.1.1.1.ACTIVITÉ DÉDUITE DE L'EXIGENCE N°1 :

Le personnel d'exploitation doit pouvoir contrôler en temps réel l'état courant de la cuve sur l'écran de son IHM.



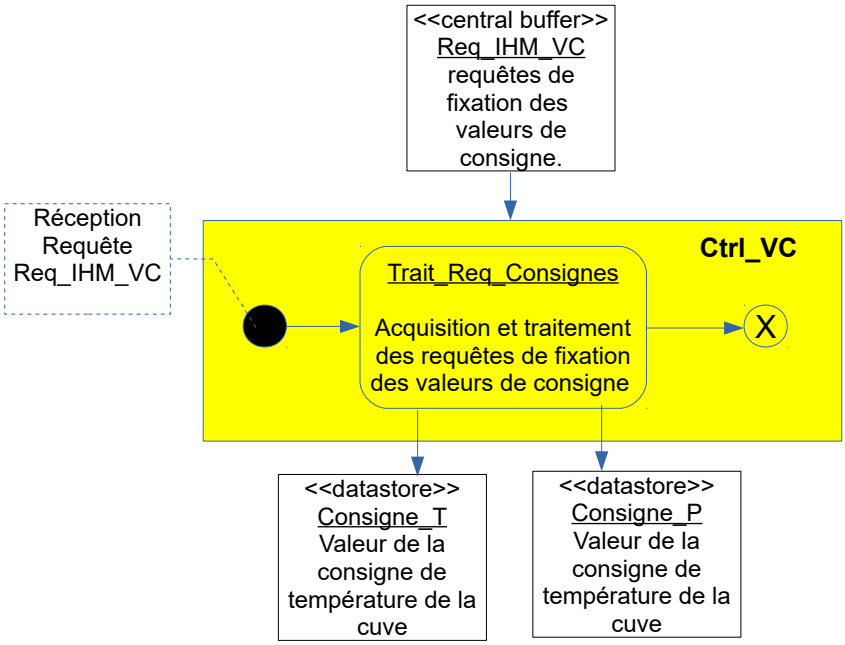
III.2.1.1.2.ACTIVITÉS DÉDUITES DE L'EXIGENCE N°2 :

ÉNONCÉ : Le personnel d'exploitation doit pouvoir archiver sur un support permanent les températures et pressions acquises ainsi que leurs dates d'acquisition.



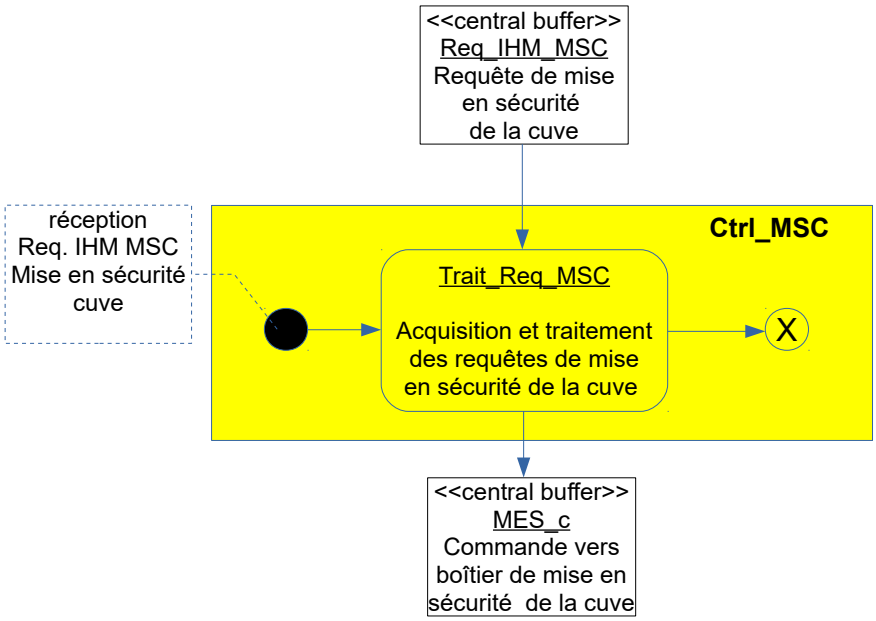
III.2.1.1.3.ACTIVITÉ DÉDUITE DE L'EXIGENCE N°3:

Le personnel d'exploitation doit pouvoir définir et contrôler les valeurs de consigne de la température et de la pression de la cuve.



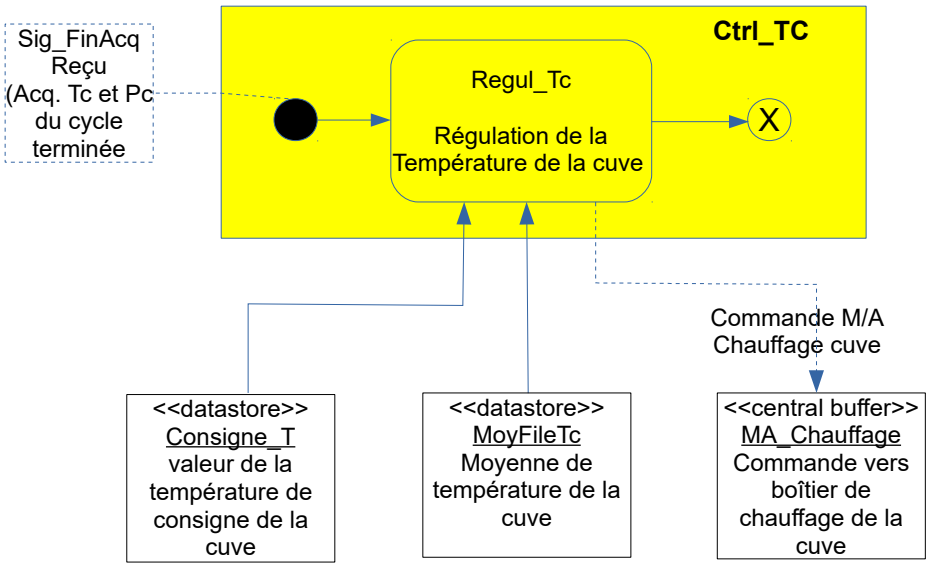
III.2.1.1.4.ACTIVITÉ DÉDUITE DE L'EXIGENCE N°4:

Le personnel d'exploitation doit pouvoir à tout moment mettre en sécurité la cuve.



III.2.1.1.5.ACTIVITÉ DÉDUITE DE L'EXIGENCE N°5:

Le système doit maintenir automatiquement la température de la cuve au voisinage de la température de consigne.



III.2.1.2.SYNTHÈSE DE L'ÉTUDE DES ACTIVITÉS :

III.2.1.2.1.ACTIVITÉS IDENTIFIÉES :

NOM	CARACTÉRISATION	CRITÈRES ET CONTRAINTES
Ctrl_Etat_Cuve	Affichage en temps réel des températures et pressions internes de la cuve sur l'écran de l'I.H.M des exploitants.	Fréquence de rafraîchissement de l'affichage = 10 hz
Archivage	Archivage des 10 dernières températures et pressions acquises ainsi de le leurs dates d'acquisition	Fréquence d'enregistrement = 1 Hz
Activité Cmd_Arch	Commande de l'archivage (marche/arrêt)	
IHM	Interface exploitant-machine : - Visualisations : mode(M/Adj), consignes, alarmes ; - Commandes M/A archivage, MES cuve.	Délais d'affichage Tc et Pc < 100 ms
Ctrl_VC	Fixation des valeurs de consigne de T et P	
Ctrl_MSC	Commande de la mise en sécurité de la cuve	Durée entre prise en compte de la commande IHM et réception commande MES par le boîtier < 100 ms
Ctrl_TC	Contrôle et régulation de la température de consigne de la cuve	Durée entre détection écart et commande de correction < 50 ms

III.2.1.2.2.FLUX DE DONNÉES, COMMANDES ET SIGNAUX IDENTIFIÉS :

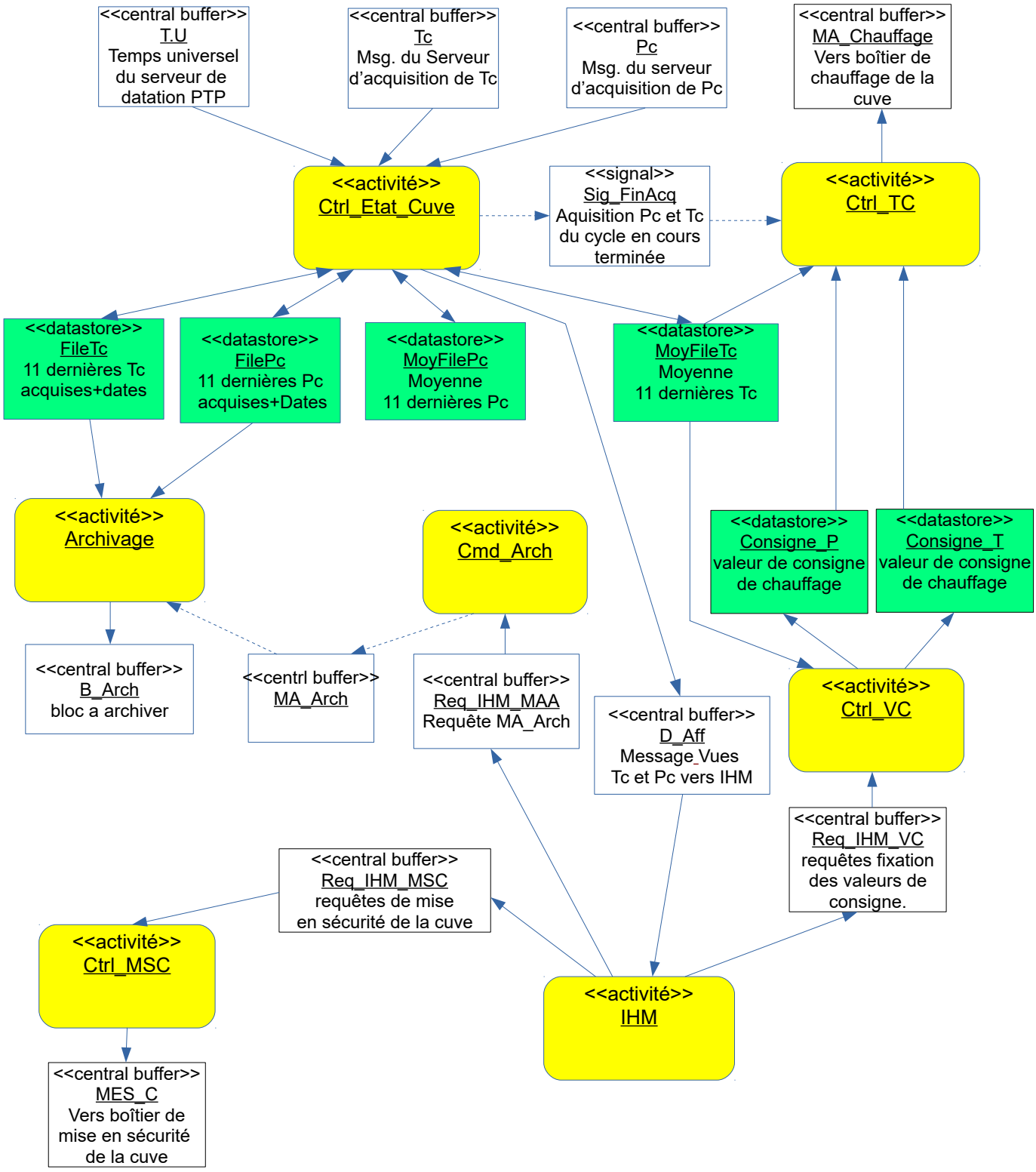
Central Buffer T.U	Message	Temps universel acquis par le serveur PTP	Serveur de datation PTP	- Ctrl_Etat_Cuve	Fréquence : 10 Hz
Central Buffer Tc	Message	Température courante mesurée par le serveur d'acquisition de la température cuve	Serveur d'acquisition de la température courante de la cuve	- Ctrl_Etat_Cuve	Fréquence : 10 hz
Central Buffer Pc	Message	Pression courante mesurée par le serveur d'acquisition de la température cuve	Serveur d'acquisition de la pression courante de la cuve	- Ctrl_Etat_Cuve	Fréquence : 10 hz
Signal Sig_FinAcq	Signal	Indique de les mesures Tc et Pc du cycle en cours ont été acquises	Ctrl_Etat_Cuve	- Ctrl_TC	Fréquence : 10 Hz
Central buffer D_Aff	Message	Vues (quadrants) Pc et Tc à afficher sur l'IHM	Ctrl_Etat_Cuve	- IHM	Vues Température et pression cuve courantes pour affichage sur l'IHM - Fréquence d'affichage = 10 Hz
Central buffer MA_Arch	Message	Commande marche/arrêt Chauffage cuve	Cmd_Arch	- Archivage	Commande alternative marche/arrêt
Central buffer B_Arch	Message	Commande d'archivage du bloc d'archivage courant	Archivage	Dispositif d'archivage	Fréquence = 10 Hz
Central buffer Req_IHM_MAA	Message	Requête IHM marche/arrêt archivage	IHM	Cmd_Arch	
Central buffer Req_IHM_VC	Message	Requête IHM de fixation des valeurs de consigne	IHM	Ctrl_VC	

Central buffer MA_Arch	Message	Commande marche/arrêt Chauffage cuve	Cmd_Arch	- Archivage	Commande alternative marche/arrêt
Central buffer Req_IHM_MSC	Message	Requête IHM de fixation des valeurs de consigne	IHM	Ctrl_MSC	
Central buffer MA_Chauffage	Message	Commande marche/arrêt du chauffage de la cuve	Ctrl_TC	Boîtier de chauffage de la cuve	
Central buffer MES_C	Message	Commande vers boîtier de mise en sécurité de la cuve	Cltrl_MSC	Boîtier de mise en sécurité de la cuve	

III.2.1.2.3.DONNÉES PERSISTANTES ET RESSOURCES COMMUNES :

ENTITÉ	CARACTÉRISTIQUES
Data store FileTc	File des 11 dernières températures cuves reçues datées
Data store FilePc	File des 11 dernières pressions cuve reçues datées
Data store MoyFileTc	Moyenne des 11 dernières températures cuve reçues
Data store MoyFilePc	Moyenne des 11 dernières pressions cuve reçues
Data store Consigne_T	Consigne de température
Data store Consigne_P	Consigne de pression

III.2.1.2.4.DIAGRAMME D'ACTIVITÉS GÉNÉRAL :



III.2.2.ÉTUDE DES CONFLITS D'ACCÈS AUX RESSOURCES PARTAGÉES:

L'étude des accès aux ressources partagées par les activités peut être résumée par le tableau suivant :

RESSOURCE	CARACTÉRISTIQUES	ACTIVITÉS UTILISATRICES	CONFLIT D'ACCÈS POSSIBLE EN CAS D'ACCÈS ASYNCHRONE
FileTc	File des 11 dernières températures cuves reçues datées	- Ctrl_Etat_Cuve (rw) - Archivage (r)	Possibilité de conflit entre Ctrl_Etat_Cuve et Archivage en cas d'accès simultané (incohérence de la donnée FileTc)
FilePc	File des 11 dernières pressions cuve reçues datées	- Ctrl_Etat_Cuve (rw) - Archivage (r)	Possibilité de conflit entre Ctrl_Etat_Cuve et Archivage en cas d'accès simultané (incohérence de la donnée FilePc)
MoyFileTc	Moyenne des 11 dernières températures cuve reçues	- Ctrl_Etat_Cuve (rw) - Ctrl_TC (r)	Possibilité de conflit entre Ctrl_Etat_Cuve et Ctrl_Tc (possibilité pour que Ctrl_TC utilise la moyenne du cycle précédent)
MoyFilePc	Moyenne des 11 dernières pressions cuve reçues	- Ctrl_Etat_Cuve (rw)	Aucun conflit possible : accès r et accès w synchronisés)
Consigne_T	Consigne de température	- Ctrl_VC (w) - Ctrl_TC (r)	Possibilité de conflit entre Ctrl_VC et Ctrl_TC (possibilité pour que Ctrl_TC utilise la valeur précédente de Consigne_T pendant 1 cycle)
Consigne_P	Consigne de pression	- Ctrl_VC (w)	Possibilité de conflit entre Ctrl_VC et Ctrl_TC (possibilité pour que Ctrl_TC utilise la valeur précédente de Consigne_T pendant 1 cycle)

III.2.3.ÉTUDE DES CONTRAINTES TEMPORELLES ASSOCIÉES À L'APPLICATION :

L'analyse des exigences permet d'identifier les contraintes temporelles auxquelles l'application doit satisfaire. Elles concernent les chaînes de traitement suivantes :

- L'affichage de la température et de la pression moyennes sur l'IHM (depuis la réception des valeurs T et P courantes jusqu'à la fin de l'affichage sur l'IHM) ;
- La régulation de la température de la cuve (depuis la réception de la température courante jusqu'à la réception de la commande de régulation par le boîtier de chauffage) ;
- La mise en sécurité de la cuve (depuis l'activation du bouton de mise en sécurité sur l'IHM jusqu'à la réception de la commande au boîtier MSE).

Le but de cette étude est de vérifier que la réaction de l'application aux sollicitations externes satisfait bien aux contraintes temporelles exprimées par la STB, qu'il s'agisse d'un temps de réaction maximal ou du déterminisme de la date de réponse.

Le résultat de cette étude est présenté ici sous la forme d'un tableau :

- La colonne "Fonction concernée" définit fonctionnellement la chaîne de traitements concernée ;
- La colonne "Détail de la chaîne de traitement" donne la composition du "chemin critique" de cette chaîne. Elle peut comprendre :
  - L'exécution d'activités (Exec <nom activite> (<durée propre max>, <récurrence>) ;
  - La propagation de flux de données ou de commandes : Flux <nom de flux> → entité réceptrice (activité, datastore, élément externe) (<Durée max>, <volume>, <récurrence> ;
  - Des périodes d'attente : Attente < événement> ;
- La colonne "Contraintes exprimées" indique les contraintes d'ordre temporel à respecter ;
- La colonne "Durée maxi" donne la durée maximale de la chaîne de traitement hors attentes .

Fonction concernée	Détail de la chaîne de traitement	Contraintes exprimées	Durée propre maxi
Régulation de la température de la cuve.	- Flux Tc → Ctrl_Etat_Cuve (1 ms, 4 o, 10 Hz) - Exec Ctrl_Etat_Cuv (1 µs, 10 hz) ; - Flux FileTc → Datastore FileTc (0,88 o, 10 hz) - Attente déclenchement Ctrl_TC - Flux FileTc_2 → Ctrl_TC ((1 µs, 88 o, 10 Hz) - Exec Ctrl_TC (1 µs, 10 hz) ; - Flux MA_Chauffage → Cuve (1 ms, 4 o, aléatoire)	- Durée max < 50 ms - La donnée FileTc doit contenir la dernière mesure Tc disponible.	< 3 ms
Affichage T moyenne sur IHM	- Flux Tc → Ctrl_Etat_Cuve (1 ms, 4o, 10 hz) - Exec Ctrl_Etat_Cuve(1 µs, 10 hz) ; - Flux D_Aff → IHM (3 ms, 300 o, 10 hz) - Exec IHM (10 ms, 100 ko, 10 Hz)	- Durée <100 ms - La donnée FileTc doit contenir la dernière mesure Tc disponible.	< 14 ms
Affichage P moyenne sur IHM	- Flux Pc → Ctrl_Etat_Cuve (1 ms, 4o, 10 hz) - Exec Ctrl_Etat_Cuve (1 µs, 10 hz) ; - Flux D_Aff → IHM (3 ms, 300 o, 10 hz) - Exec IHM (10 ms, 10 Hz)	- Durée < 100 ms - La donnée FileTc doit contenir la dernière mesure Tc disponible.	< 14 ms
Mise en sécurité cuve	- Exec IHM (prise en compte interaction exploitant) (10 ms, aléatoire) - Flux Req_IHM_MSC → Ctrl_MSC (3 ms, 1ko, aléatoire) - Exec Ctrl_MSC (3 µs, aléatoire) - Flux MES_C → Boîtier MES Cuve (3 ms, 4 o, aléatoire)	Durée < 100 ms	< 16 ms

III.3.CONCEPTION DE L'ARCHITECTURE SYSTÈME :

III.3.1.ÉTUDE DE LA COUCHE PRÉSENTATION :

III.3.1.1.ANALYSE DES EXIGENCES:

III.3.1.1.1.RESPECT DES NIVEAUX DE FIABILITÉ ET MAINTENABILITÉ:

Selon les spécifications de besoins, il suffit d'un seul exploitant pour piloter le système.

Cependant, il est précisé que la probabilité d'une panne interdisant l'utilisation des fonctions IHM doit, sur une année d'exploitation, être inférieure à 10^{-2} et que les opérations de remise en état doivent être effectuées "à chaud".

REMARQUE PRÉLIMINAIRE : *un SPARE (une rechange en français) est ici un poste d'exploitation entièrement configuré des points de vue matériels et logiciels mais non connecté au système ni à l'alimentation électrique qui est maintenu en réserve pour prendre la place d'un poste défaillant.*

***ÉTUDE AVEC UN SEUL POSTE ET UN SPARE*:**

Le M.T.B.F d'une machine de classe courante peut être estimé autour de 25000 heures (environ 3 ans). La probabilité de panne d'un composant peut être estimée par la formule :

$$R(t) = 1 - e^{-t/\theta}$$

θ étant le M.T.B.F et t le temps écoulé depuis le début d'exploitation du système.

Avec t = 8760 h (un an) et θ = 25000, nous obtenons une fiabilité de :

$$1 - e^{-8760/25000} \simeq 0{,}30\ (30\%)$$

Soit une probabilité de panne sur l'année d'environ 30 %. En cas de défaillance du poste d'exploitation, le service peut être rétabli rapidement (en quelques minutes), mais la configuration ne peut satisfaire à l'exigence d'une probabilité de panne inférieure à 10^{-2} sur un an.

ÉTUDE AVEC DEUX POSTES EN POSITION REDONDANTE ET UN SPARE:

Dans ce cas, supposons que les deux postes P1 et P2 aient été démarrés à la même date et qu'un des deux postes tombe en panne à la date T (par exemple, le poste P1). Le service n'est pas interrompu, mais on passe en mode dégradé à un seul poste (P2). La probabilité de panne de P1 à la date T est :

$$P_T = 1 - e^{-T/\theta}$$

La probabilité que le poste P2 tombe en panne pendant la durée D du M.T.T.R est :

$$(1 - e^{-(T+D)/\theta}) - (1 - e^{-T/\theta}) = e^{-T/\theta} - e^{-T/\theta} * e^{-D/\theta}$$. Donc :

$$P2_{MTTR} = e^{-T/\theta} * (1 - e^{-D/\theta})$$

La probabilité pour que les deux postes tombent en panne avant la remise en état du premier poste défaillant est :

$$P2D = P_{T1} * P_{MTTR} = (1 - e^{-T/\theta}) * e^{-T/\theta} * (1 - e^{-D/\theta})$$

Remarquons que les deux premiers facteurs sont inférieurs à 1. Un majorant de P2P est donc :

$$P2P < 1 - e^{-D/\theta}$$

Supposons que dans le cas qui nous occupe la durée de remise en état du poste défaillant (M.T.T.R) soit estimée à moins de 1 heure (Ce délais est largement surestimé car cette remise en état peut se résumer au remplacement du poste défaillant par le spare). Pour le cas qui nous occupe, la probabilité pour que les deux postes tombent en panne avant la remise en état du premier poste défaillant est :

$$P2P = 1 - e^{-1/25000} = 4*10^{-5}$$

valeur qui satisfait très largement les exigences ($>10^{-2}$).

III.3.1.1.2. RESPECT DES DÉLAIS DE PRISE EN COMPTE DES COMMANDES :

Quelle que soit l'architecture choisie, les durées maximales de propagation des commandes IHM au reste de l'application (activité IHM) peuvent être estimées à quelques ms ($\simeq$3) dans le cas où le client et le serveur d'application sont sur des machines différentes connectées à un réseau dédié (quelques microsecondes dans le cas contraire). Ces durées sont négligeables par rapport aux délais de réactions exigés pour les commandes de l'IHM (<100 ms).

III.3.1.1.3.RESPECT DES DÉLAIS D'AFFICHAGE DES VUES:

- Dans le cas des solutions "client léger" ou "client riche", les données d'affichage transmises au client par le réseau représentent la description HTML de la page à afficher, soit un volume de données maximal de l'ordre de 100 ko (soit 1 mbit). La transmission de 100 ko sur un réseau TCP/IP se fait en deux paquets, ce qui, avec le mécanisme des acquittements, peut allonger considérablement la durée de transfert (quelques ms).
- Dans le cas d'une solution "client lourd" les données d'affichage transmises au client se résument à une quarantaine d'octets tous les 10 hz. Ces données peuvent

donc être transmises en une seule trame en quelques microsecondes.

Ces durée restent négligeables par rapport aux délais d'affichage exigés (<100 ms);

III.3.1.1.4.RESPECT DES REPRÉSENTATIONS GRAPHIQUES EXIGÉES :

Les graphismes demandés pour l'affichage de la température et de la pression courante sont impossibles à obtenir pour un client léger. Ils pourraient être obtenus par un "client riche" utilisant (par exemple) des applets java, ou un "client lourd".

III.3.1.2.SOLUTION CHOISIE :

Compte tenu :

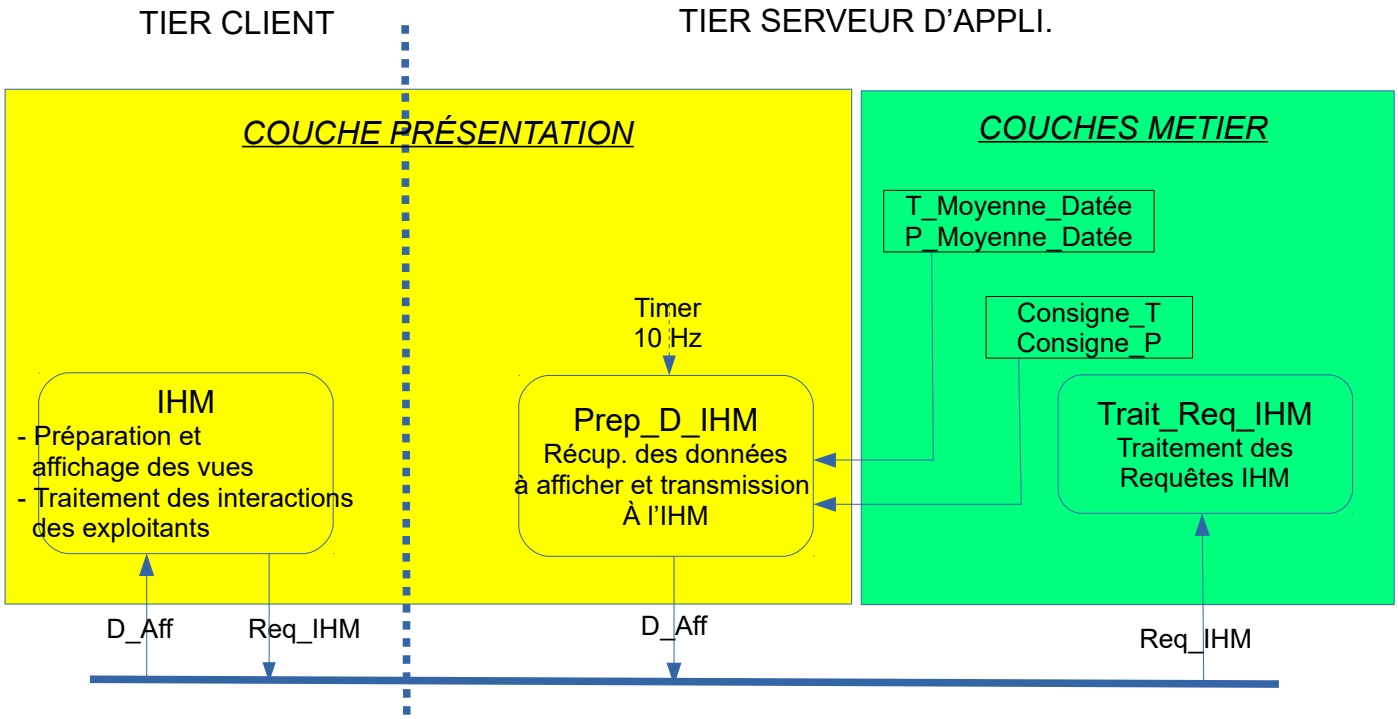
- Des représentations graphiques à réaliser ;
- Des exigences en matière de disponibilité et de délais de remise en état ;
- De l'exigence de remplacement "à chaud" d'un poste d'exploitation défaillant ;
- Du faible nombre de postes d'exploitation et des faibles besoins d'évolution de ce nombre ;
- De la localisation des postes dans un local sécurisé situé dans l'enceinte de l'implantation ;
- Du fait que l'exploitation est réservée à un faible nombre d'individus bien identifiés.

Une solution "client lourd" avec deux postes d'exploitation redondants paraît la plus indiquée. Ces postes d'exploitation seront équipés d'un logiciel d'IHM graphique développé à partir d'un environnement de développement intégré (tel que DELPHI, par exemple). Ils seront connectés au reste de l'application par un réseau local dédié.

La redondance des postes sera obtenue par l'organisation suivante :

- Un poste "maître" permet de commander le processus et de visualiser les états ;
- Un poste "adjoint" se contente de visualiser les informations d'état ;
- En cas de dysfonctionnement du maître, la promotion du poste adjoint en poste maître ne nécessite que l'appui sur un bouton de son IHM.
- L'activité Prep_D_IHM sera localisée dans le tier serveur d'application. Toutes les 100 ms, cette activité prélèvera dans les puits de données T_Moyenne_Datée, P_Moyenne_Datée, Consigne_T et Consignes_P les données à afficher sur l'IHM et les transmettra à l'activité Aff_IHM supportée par les deux postes redondants (en mode UDP multicast) ;
- L'activité Aff_IHM, située dans les deux machines du tier client, sera déclenchée par la réception du message réseau D_Aff. Elle sera chargée de construire et/ d'afficher la page de l'IHM à partir des données reçues ainsi que d'émettre vers l'activité Trait_Req_IHM les commandes saisies par les exploitants.

Le schéma suivant donne la solution de répartition de la couche présentation :



III.3.2.ÉTUDE DES COUCHES MÉTIER :

III.3.2.1.ANALYSE DES EXIGENCES:

III.3.2.1.1.ÉVALUATION DE LA PUISSANCE DE TRAITEMENT NÉCESSAIRE :

Le tableau suivant recense les "chemins critiques" des traitements assurés par l'application :

Traitements	**Chemins critiques :** Activités concernées, durées propres max., mémoire occupée, fréquence.	**Flux concernés volumes utiles et fréquences** (flux externes en gras)	**Contraintes de durée (ms)**
Régulation de la température de la cuve.	- Ctrl_Etat_Cuve (1 ms, 10 ko, 10 hz) - Ctrl_TC (1 ms, 10 ko, 10 hz)	- **Tc** (4 o, 10 Hz) - MoyFileTc (8 o, 10 Hz) - **M/A_Chauffage** (4 o, Aléatoire.)	< 50
Marche/arrêt de l'archivage ;	- **IHM** (10 ms, 10 ko, aléatoire) - Cmd_Arch (1 µs, 10 ko, aléatoire)	- **Interactions_IHM** (16 o, aleatoire) - Requête Req_IHM_MAA (1000 o, aléatoire) - MA_Arch (4 o, Aléat.)	< 100
Mise en sécurité de la cuve;	- **IHM** (10 ms, 10 ko, aléatoire) - Ctrl_MSC (1 µs, 10 ko, aléatoire)	- **Interactions_IHM** ((16 o, Aleatoire) - Req_IHM_MSC (1000 o, aléatoire) - **MES_C** (4 o, Aleat)	< 100
Acquisition des valeurs des paramètres de consigne (température de consigne et pression maximale admissible).	- **IHM** (10 µs, 10 ko, aléatoire) - Crtl_MSC (1 µs, 10 ko, aléatoire)	- **Interactions_IHM** (aléatoire.) - Req_IHM_VC (1000 o, aléatoire) - Consignes (4 o, Aleat)	Non défini (à priori, <3000)
Archivage	- Ctrl_Etat_Cuve (1 ms, 10 ko, 10 Hz) - Archivage (1 ms, 10 ko, 1 Hz)	- **Tc** (4 o, 10 Hz) - **Pc** (4 o, 10 Hz) - **B_Arch** (192 o, 1 Hz)	Non défini (à priori <1000)
Affichage des données sur l'IHM	- Ctrl_Etat_Cuve (1 ms, 10 ko, 10 Hz) - **IHM** (10 ms, 100 ko, 10 Hz)	- Tc (4 o, 10 Hz) - Pc (4 o, 10 Hz) - **Vues** (16 o, 10 Hz)	100

REMARQUES : Les activités exécutables par le tiers serveur d'application et les flux externes sont notés en gras.

Nous pouvons constater que la durée de ces chemins critiques respecte très largement les contraintes de durée qui sont imposées par les spécifications.

D'autre part, concernant les activités "métier", le pourcentage d'occupation maximal d'une unité de traitement de type intel I5 induit par l'ensemble des activités peut être estimée, en régime constant, à :

(1*10+1*10+1*10+1*10+1)/1000 = 4,1%.

Il semble donc qu'une seule machine de traitement de puissance voisine d'un intel I5 suffise pour étaler la charge dela couche métiers.

III.3.2.1.2.PRISE EN COMPTE DES EXIGENCES DE FIABILITÉ :

Les exigences de fiabilité concernant la régulation de la température de la cuve :

- Probabilité de panne sur une année d'exploitation: 10^{-4}
- Remise en état à chaud.

ne peuvent être atteintes avec une seule machine de type courant dans le TIER serveur d'application. En effet :

- D'une part, la remise en état "à chaud" ne peut être envisagée si l'étage serveur d'application est supporté par une seule machine ;
- D'autre part, le calcul de fiabilité effectué lors de l'étude de fiabilité de l'étage client peut être repris ici : pour atteindre une probabilité de panne < 10^{-4}, il faut choisir au moins une architecture avec deux machines redondantes.

III.3.2.2.SOLUTION CHOISIE :

III.3.2.2.1.PRINCIPE :

Cette solution est basée sur une grappe de deux machines redondantes connectées à un réseau local. Une seule des machines (le MAÎTRE) commande le processus. L'autre machine (L'ADJOINT) effectue également tous les traitements, mais ne fournit pas ses données à la périphérie.

Il faut également disposer d'un "spare" prêt à être branché à la place d'une machine défaillante.

Chacune des machines surveille le fonctionnement de l'autre par l'intermédiaire de messages réseaux échangés entre elles. Un mécanisme implanté dans les deux machines réalise les traitements suivants :

- Si le maître tombe, l'adjointe prend le relais automatiquement. La machine en panne peut alors être remplacée par le spare et l'on revient au fonctionnement nominal. La disponibilité du système n'a pas été affectée ;
- Si l'adjointe tombe, le maître continue à assurer le service mais signale l'événement aux exploitants. La fiabilité est dégradée pendant le remplacement de l'adjointe par le spare, puis on revient au fonctionnement nominal. Là aussi, la disponibilité du système n'a pas été affectée.

NOTA : pour que le basculement adjoint → maître soit sans conséquence du point de vue de l'exécution, il doit s'effectuer en moins de 100 ms.

REMARQUES

- Si après basculement de l'adjointe en maître, ce nouveau maître tombe en panne avant que le spare n'ait été mis en place, le service se trouve interrompu. C'est à la survenue de cet enchaînement de pannes que s'applique la probabilité < 10^{-5}.
- Comme pour l'étage client, la durée maximale de la remise en état est fixée à une heure. Dans ces conditions, la probabilité de panne du système est inférieure à 4*10-5 (voir étude de la couche présentation), ce qui satisfait le critère de fiabilité demandé.

III.3.2.2.2.FONCTIONNEMENT DÉTAILLÉ:

FONCTIONNEMENT NOMINAL :

Considérons l'installation en fonctionnement nominal : les deux machines du tier serveur d'application sont en fonction. L'une d'elles joue le rôle de maître, l'autre le rôle d'adjointe.

Les deux machines acquièrent les données de la périphérie (P, T et commandes utilisateurs) et exécutent toutes les activités dévolues au tier serveur d'application. Elles archivent également les données acquises sur deux supports différents et élaborent le message Daff qu'elles émettent sur le réseau à destination des deux postes d'exploitation. En revanche, seule la machine maître émet vers les Boîtiers MES et Chauffage).

Chacune des deux machines émet vers l'autre machine un message d'interrogation destiné à tester le fonctionnement de cette dernière. La machine réceptrice répond alors en transmettant à l'émettrice un message d'acquittement.

DÉTECTION D'UN DYSFONCTIONNEMENT DE L'ADJOINTE :

En cas de réponse erronée ou d'absence de réponse de l'adjointe au message d'interrogation du maître après un certain délais, le maître signale ce dysfonctionnement aux exploitants par l'intermédiaire de son I.H.M (message écrit, signal sonore…).

DÉTECTION D'UN DYSFONCTIONNEMENT DU MAÎTRE :

En cas de réponse erronée ou d'absence de réponse du maître au message d'interrogation de l'adjointe après un certain délais, la machine adjointe passe en position de maître et signale ce dysfonctionnement aux exploitants par l'intermédiaire de son I.H.M (message écrit, signal sonore…).

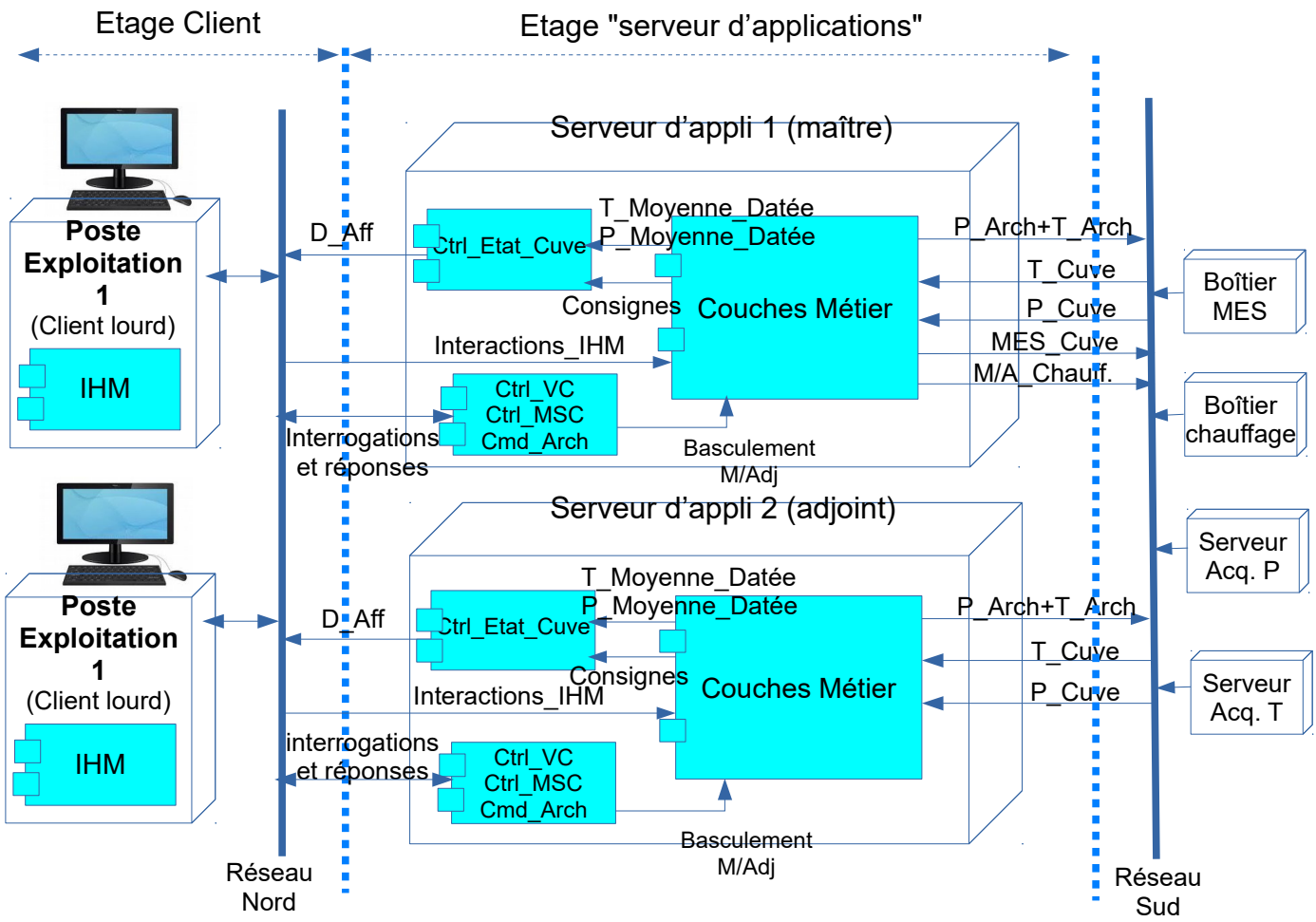
INITIALISATION DES MACHINES AU LANCEMENT DES APPLICATIONS:

Par défaut, l'application est lancée sur une machine en mode ADJOINTE. Elle émet alors sur le réseau un message d'interrogation. Deux cas peuvent alors se présenter :

- L'autre machine n'est pas encore lancée ou est en panne. L'interrogation échoue et l'application passe en mode maître ;
- Une autre machine est déjà lancée : cette autre machine étant forcément en mode maître, la machine nouvellement lancée reste en mode ADJOINTE.

III.3.2.2.3. DIAGRAMME DE DÉPLOIEMENT DE L'ÉTAGE SERVEUR D'APPLICATIONS :

Compte tenu de ce qui précède, le diagramme ci-dessous représente une proposition pour l'architecture et le déploiement de l'étage "serveur d'applications" :

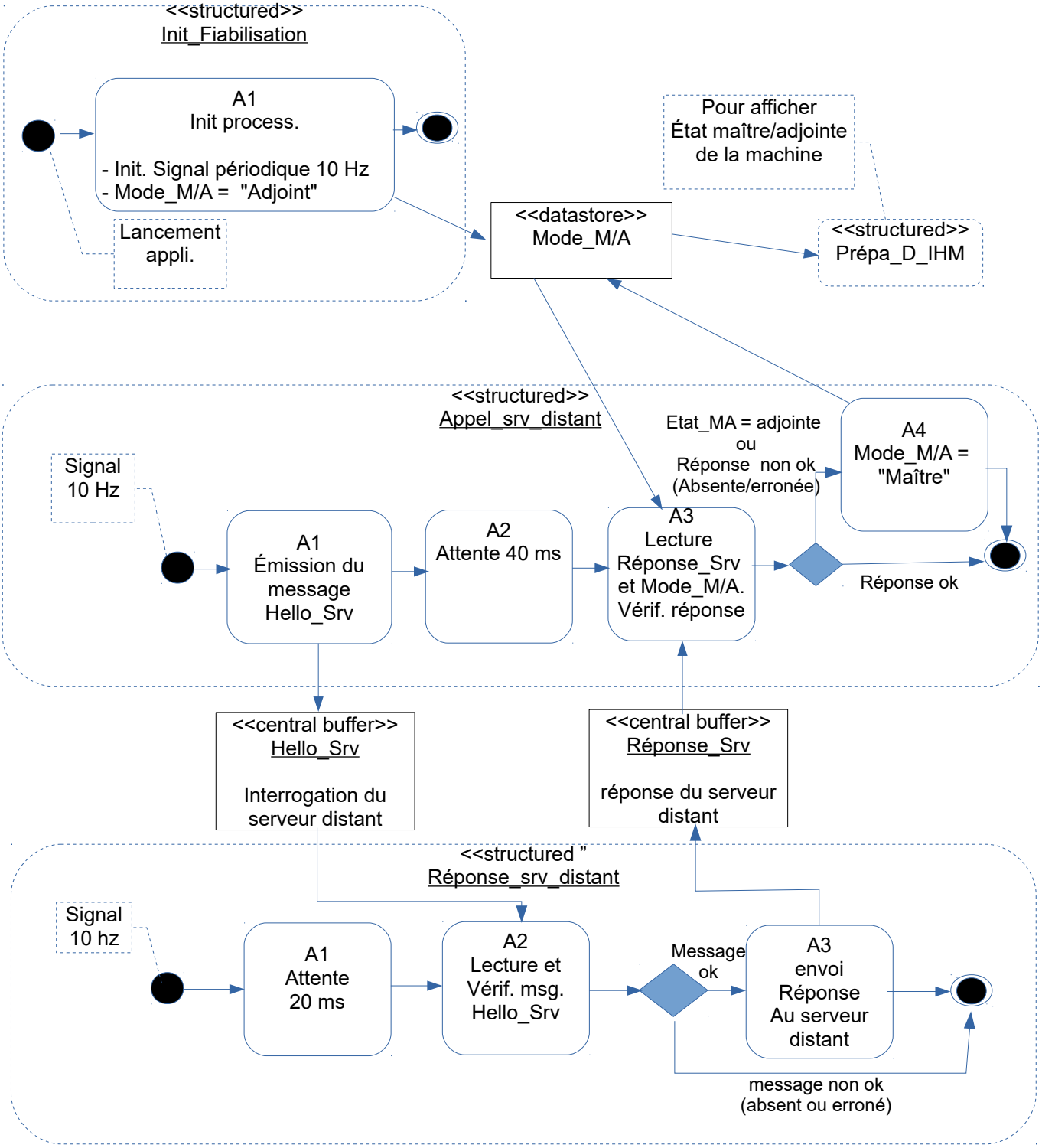


COMMENTAIRES :

- Le réseau "nord" est utilisé pour la communication entre les logiciels des postes d'exploitation (clients lourds) et le reste de l'application et pour la communication entre les machines de l'étage serveur d'applications liée à la gestion des rôles maîtres et adjoint ;
- Le réseau "sud" est dédié :
  - A l'acquisition des mesures T et P ;
  - A la commande des boîtiers de régulation du chauffage et à la mise en sécurité de la cuve ;
  - A la communication avec le tier "serveur de données" .
- L'ensemble des flux échangés sur les deux réseaux et les contraintes liées à ces flux peuvent largement être assurés par des réseaux locaux sur paire torsadée (100 mbytes) ;
- Le choix de deux réseaux séparés au lieu d'un réseau unique est justifié uniquement par le soucis de dissocier les échanges principalement asynchrones liés aux interactions opérateurs des échanges synchrones liés au traitement des mesures qui sont soumis à des contraintes de délais.

III.3.3.FIABILISATION :

Cette activité peut être décomposée en trois sous-activités : Init_Fiabilisation, Appel_srv_distant et Réponse_à_srv_distant :



REMARQUES :

- La ressource partagée Mode_M/A (type booléen) est ajoutée. Elle mémorise le mode de fonctionnement de la machine locale (maître/adjointe) ;
- Action A1 de Init_fiabilisation: au lancement de l'activité, le mode de fonctionnement est initialisé à "adjointe" ;
- A chaque événement 10 Hz, l'activité Appel_srv_distant est déclenchée. Elle émet immédiatement vers le serveur distant un message Hello_Srv, puis attend 40 ms pour aller récupérer le message Réponse_Srv du serveur distant.
  - Si ce message est présent et correct, l'activité est stoppée.
  - Sinon, si la machine locale est en mode adjointe et le message Réponse_Srv absent ou incorrect, elle passe en mode maître en modifiant la ressource Mode_M/A ;
  - Sinon (machine en mode maître), l'activité est stoppée.
- A chaque événement 10 Hz, l'activité Réponse_srv_distant est déclenchée. Elle attend alors 20 ms avant d'essayer de lire le message Hello_Srv émis par le serveur serveur distant.
  - Si ce message est présent et correct, l'activité renvoie le message Réponse_Srv au serveur distant, puis elle est stoppée.
  - Sinon, l'activité est stoppée.

Sous_activité Init_Fiabilisation	
DOMAINE	**OBSERVATIONS**
Récurrence	Démarrée uniquement au lancement de l'application.
Durée propre	< 100 ns.
Concurrence sur ressource Mode_M/A	Avec Prépa_D_IHM et Appelle_Srv distant : aucune car ces deux activités ne sont pas encore démarrées quant l'action A1 de Init_Fiabilisation s'exécute.
Sous_activité Appelle_srv_distant	
DOMAINE	**OBSERVATIONS**
Récurrence	10 Hz
Durée propre	< 1 µs
Concurrence sur ressource Mode_M/A	- Avec Init_Fiabilisation : aucune (activité non lancée lors de l'init) ; - Avec Prépa_D_IHM : importante (les deux activités avec récurrence 10 hz), mais sans danger (donnée Mode_M/A atomique)
Sous_activité Répond_à_srv_distant	
DOMAINE	**OBSERVATIONS**
Récurrence	10 Hz
Durée propre	< 1 µs

III.3.4.ÉTUDE DE LA COUCHE DONNÉES :

III.3.4.1.ANALYSE DES EXIGENCES :

Le besoin exprimé est l'archivage sur un support permanent, à la fréquente de 1 hz, des 10 dernières températures et pressions acquises ainsi que leurs dates d'acquisition. Le flux des données utiles à archiver est de 120 octets/s, soit environ 10,4 Mbytes pour un fonctionnement continu de 24 heures. Ces exigences sont évidemment peu dimensionnantes. Cependant, les contraintes imposées :

1. Le dysfonctionnement d'une unité de stockage ne doit pas occasionner de pertes des mesures acquises pendant la remise en état ;
2. Possibilité de remplacer une unité de stockage à chaud ;
3. Taux de défaillance annuel (Annualized Failure Rate) inférieur à 10^{-2}.

Sont impossibles à satisfaire avec une seule unité de stockage en attachement direct .

En effet, la contrainte n°1 implique qu'il existe au moins deux unités de stockage indépendantes et effectuant toutes deux simultanément l'archivage des données .

Un attachement direct de chacune de ces unités sur les deux serveurs d'application redondants proposés pour l'étage serveur d'applications ne peut également pas convenir car la contrainte n°2 impose un remplacement à chaud.

III.3.4.2.ÉTUDE DES SOLUTIONS :

III.3.4.2.1.UTILISATION D'UN N.A.S AVEC GESTION EN RAID 5 :

Considérons la solution basée sur un NAS (Network Attachment System) équipé de trois unités de stockage gérées par un RAID de niveau 5 et acceptant le remplacement à chaud d'une unité de stockage et la régénération automatique du contenu de celle-ci en cas de remplacement à chaud d'une unité.
Cette solution, qui permet de satisfaire aux exigences n° 1 et 2, a l'avantage de la simplicité. Cependant, elle ne satisfait pas à la contrainte de fiabilité. En effet, le M.T.B.F d'un N.A.S de moyenne gamme est de l'ordre de 10 ans (876000 h). Pour cette valeur, le taux de défaillance annuel est de :

$$A.F.R = 1\text{-}e^{-8760/876000} \simeq 0{,}095$$

Ce qui est notoirement insuffisant. En fait, il faudrait un M.T.B.F de l'ordre d'une centaine d'année pour y parvenir. Certains NAS annoncent ce niveau de performances, mais il s'agit de matériels très onéreux, d'autant plus que ces niveaux ne sont basés que sur des projections statistiques et qu'il faudra posséder également un spare pour le dépannage.

III.3.4.2.2.UTILISATION DE DEUX DISQUES DURS CONNECTABLES PAR RÉSEAU :

Cette deuxième solution consiste à utiliser un cluster de trois disques durs connectés aux serveurs d'application par l'intermédiaire d'un réseau ethernet.

Chacun des deux serveurs d'applications redondants a accès à ce cluster de disques par l'intermédiaire d'un raid logiciel de niveau 5, mais seul le serveur "maître" écrit effectivement sur cet espace mutualisé.

L'activité "archivage" est exécutée simultanément dans chacun des deux serveurs d'application, mais n'archive effectivement que dans le serveur maître.

COMPORTEMENT EN CAS DE PANNE D'UN SERVEUR D'APPLICATION :

- Une panne du serveur adjoint est sans conséquence pour l'archivage ;
- En cas de panne du serveur maître, le serveur adjoint est promu en maître en moins de 100 ms et prend le relais de l'archivage qui, de ce fait, n'est pas affecté par la panne .

COMPORTEMENT EN CAS DE PANNE D'UN DISQUE DUR :

La panne d'un des disques durs d'archivage est sans conséquence sur les données archivées à condition que ce disque dur soit changé avant qu'une panne deuxième panne survienne sur un des disques restants. L'attachement par réseau permet un changement à chaud. Le RAID 5 permet la reconstitution des données sur le disque de rechange .

PROBABILITÉ DE PANNE ANNUELLE DU DISPOSITIF:

Une probabilité de panne annuelle "réaliste" d'un disque dur peut être estimée à 8 %, ce qui correspond à un MTBF de 100000 heures environ (11 ans). Supposons que dans notre dispositif, le remplacement d'un des disques disque dur par un spare dure au maximum 1 heure. D'après ce qui précède, le dispositif n'est hors service qu'en cas de survenue d'une panne de deux disques en moins de une heure. La probabilité d'un tel enchaînement de panne sur une année est inférieure à :

$$P2P = 1 - e^{-1/100000} = 0.00000999995 = 10^{-5}$$

Le dispositif satisfait donc au critère de fiabilité imposé par la Spécification des Besoins.

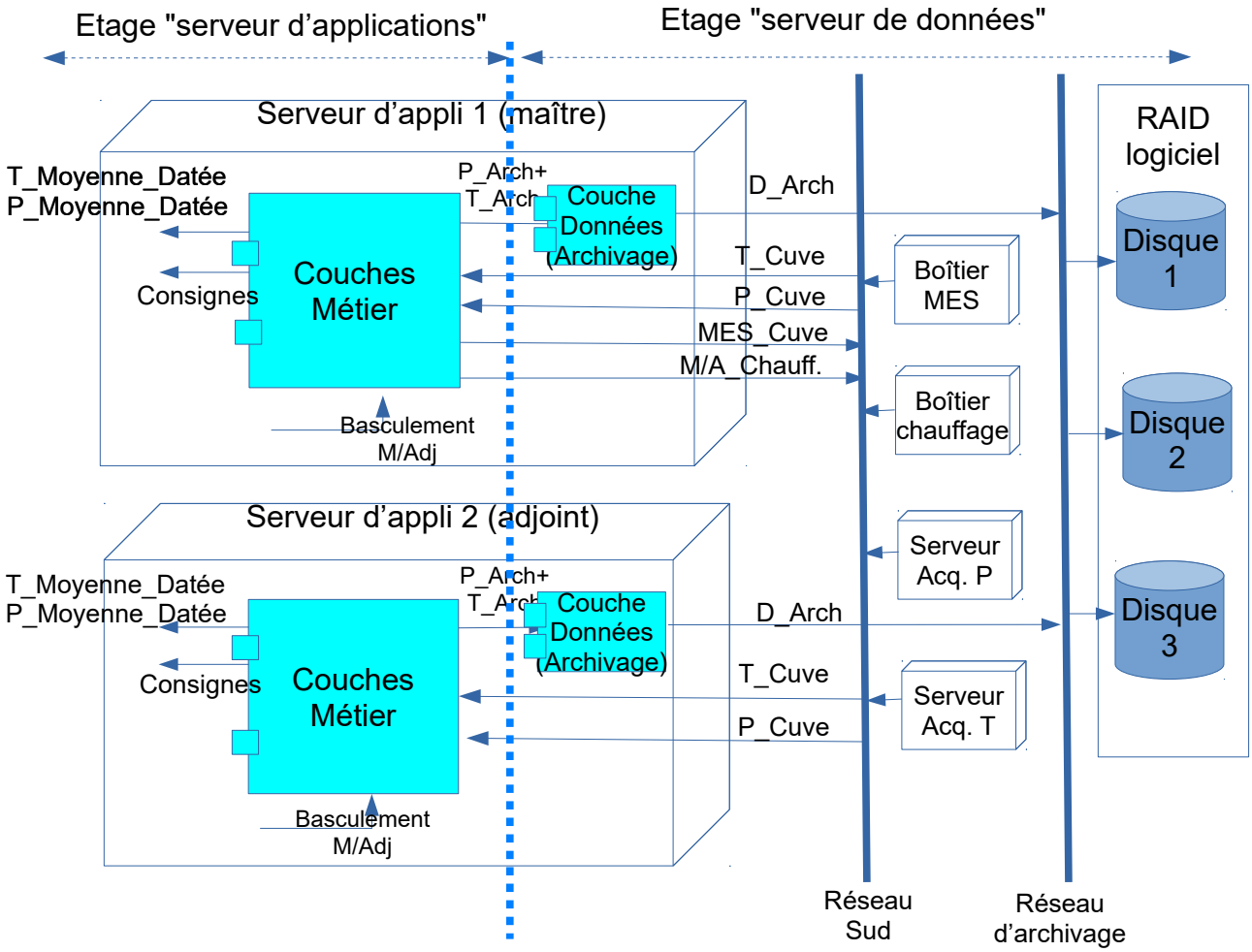
RISQUE DE CONGESTION DU RÉSEAU SUD :

On pourrait être tenté d'utiliser le réseau "sud" pour connecter le cluster de disques durs. Cependant, si la coexistence des messages liés à l'acquisition des mesures de pression et de température et des commandes du processus (MES et MA_chauffage) avec les messages d'archivage ne pose à priori pas de problème, il n'en est pas de même des messages engendrés par la gestion des RAID.

En effet, les messages d'archivage sont relativement courts (120 octets utiles). De plus, ces messages peuvent être synchronisés avec le cycle d'acquisition de 100 ms. En revanche, les messages de gestion du RAID (recopie des données, régénération en cas de panne d'un disque dur) peuvent être très volumineux. De plus, ils sont totalement asynchrones du cycle de 100 ms des acquisitions.
Il semble donc raisonnable de connecter le cluster de disques sur un réseau indépendant du réseau "sud".

III.3.4.3.SOLUTION CHOISIE :

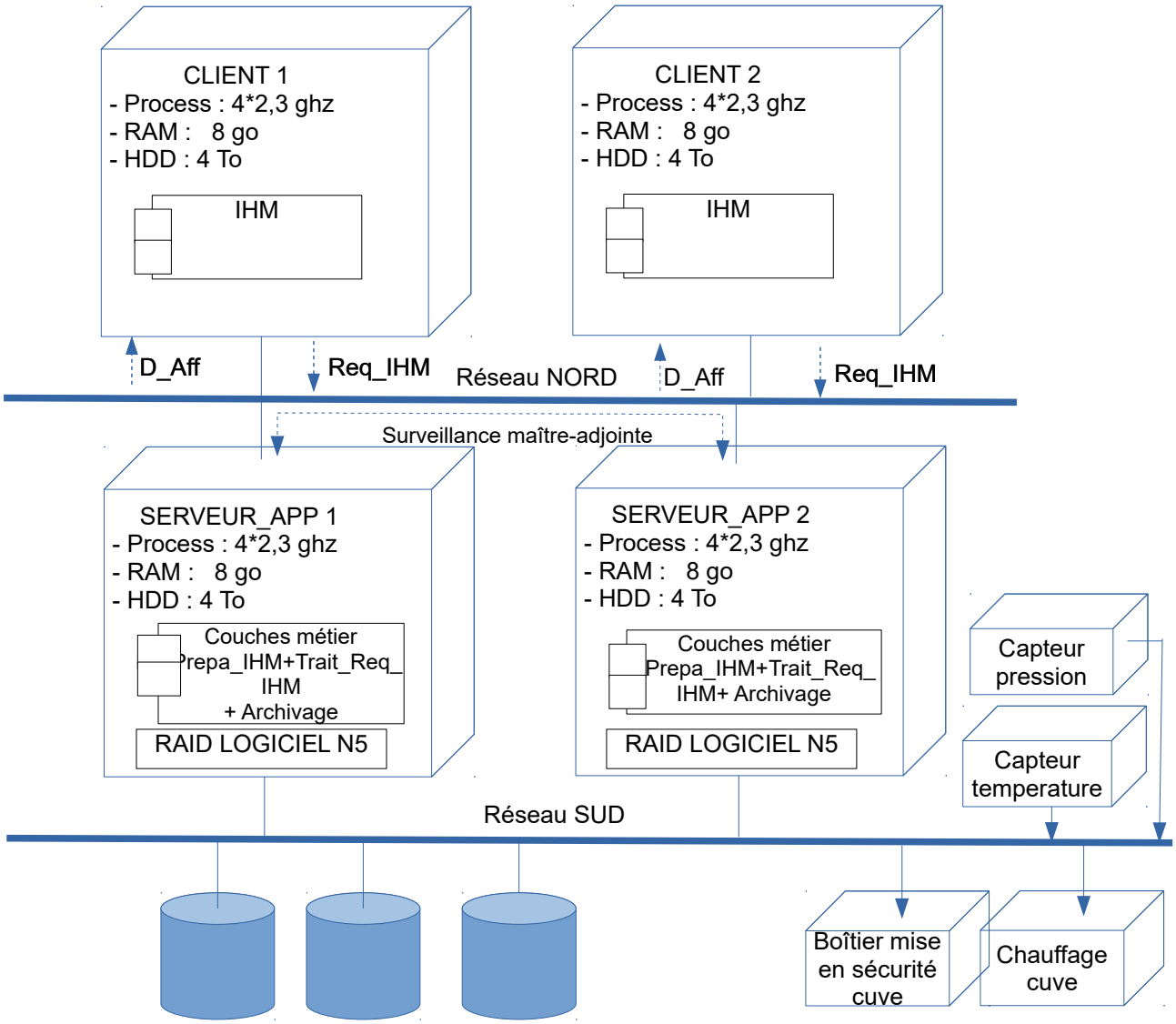
La solution proposée est donc la deuxième (utilisation de deux disques durs connectables aux serveurs d'applications par un réseau dédié et gérés par des raids de niveau 5). Cette solution correspond au diagramme de déploiement suivant :



REMARQUE : Dans cette solution, l'activité Archivage (préparation de l'archivage) est localisée dans les deux serveurs de l'étage serveur d'application.

III.3.5.SOLUTION GLOBALE ET PROPOSITION TECHNIQUE :

Le diagramme de déploiement ci-dessous représente la synthèse des études architecturales et fonctionnelles effectuées précédemment pour les couches PRÉSENTATION, MÉTIER et DONNÉES. Il pourra servir de base pour la rédaction d'une proposition technique et commerciale.



Les principales options architecturales choisies y sont représentées :

- Un étage CLIENT dont la partie affichage IHM et saisie des commandes opérateurs repose sur deux postes d'exploitation redondants connectés aux serveurs d'application par un réseau (réseau "nord"). Le réseau nord est utilisé également pour transmettre les messages de surveillance mutuelle entre serveurs

d’application;

- Un étage SERVEUR D’APPLICATIONS reposant sur deux machines redondantes (maître et adjointe) hébergeant les activités de la couche métier, mais également les activités "préparation de l’affichage" et "préparation de l’archivage". Ces deux serveurs sont connectés aux différents équipements d’acquisition et de contrôle liés à la cuve par l’intermédiaire d’un deuxième réseau, le réseau "sud". Le fonctionnement est le suivant :
  - Maître et esclave se surveillent mutuellement par l’intermédiaire de messages d’interrogation et de réponse circulant sur le réseau nord ;
  - Seul le maître assure les fonctions d’affichage et d’archivage ;
  - Le basculement maître/adjoint est déclenché automatiquement par le dialogue de surveillance entre les deux machines.
- L’étage SERVEUR DE DONNÉES repose sur un cluster de 3 disques connectés aux serveurs d’application par un réseau dédié uniquement à ces échanges et aux échanges liés au RAID (le réseau d’archivage). Le cluster est accessible par les deux serveurs à travers un RAID LOGICIEL de niveau 5. A un instant donné, seule la machine en position de maître écrit dans cet espace.

III.4.CONCEPTION LOGIQUE :

III.4.1.RAPPELS :

III.4.1.1.OBJET DE L'ÉTUDE :

La conception logique a pour objectif de construire l'architecture logique (ou statique) de l'application. Cette architecture se présente sous la forme d'un graphe dont les nœuds sont les composants logiques de l'application (modules logiciels) et dont les arcs représentent les relations "statiques" entre ces composants.

La méthode d'analyse qui a été choisie ici est la CONCEPTION PAR OBJETS. A dessein, la spécification des besoins n'a pas utilisé la symbolique U.M.L des cas d'utilisation pour spécifier les exigences. Or, pour démarrer une conception par objets, cette symbolique est souvent plus pratique. Nous allons donc traduire les EXIGENCES exprimées dans la spécification des besoins en cas d'utilisation, puis rechercher pour chaque cas les CATÉGORIES qui concourent à sa réalisation. La maîtrise de cet exercice est souvent utile lorsque l'on ne dispose pas de documents de spécification basés sur U.M.L.

Le résultat de la conception par objet est représenté par le DIAGRAMME DE CLASSES de l'application. Pour construire ce diagramme, nous allons suivre la démarche suivante :

POUR (chaque exigence exprimée par la STB de l'appli.) FAIRE
- Exprimer cette exigence sous la forme d'un CAS D'UTILISATION ;
- En déduire les CATÉGORIES collaborant à ce cas, les attributs des ces catégories et les méthodes des classes correspondant à ces catégories ;

FIN FAIRE
A partir de ces résultats, construire le diagramme de classes de l'application.

III.4.1.2.LISTE DES EXIGENCES :

Les exigences exprimées par la S.T.B sont les suivantes :

- Le PERSONNEL D'EXPLOITATION doit pouvoir CONTRÔLER EN TEMPS RÉEL l'ÉTAT COURANT DE LA CUVE sur l'écran de son IHM.
- Le PERSONNEL D'EXPLOITATION doit pouvoir ARCHIVER sur un support permanent les TEMPÉRATURES et PRESSIONS acquises ainsi que leurs DATES D'ACQUISITION ;
- Le PERSONNEL D'EXPLOITATION doit pouvoir DÉFINIR les VALEURS DE CONSIGNE de la température et de la pression de la cuve ;
- L'APPLICATION doit MAINTENIR AUTOMATIQUEMENT la TEMPÉRATURE DE LA CUVE au voisinage de la TEMPÉRATURE DE CONSIGNE.

Chacune de ces exigences peut être assimilée à un CAS D'UTILISATION.

III.4.2.ANALYSE DES CAS D'UTILISATION :

III.4.2.1.CAS D'UTILISATION "CONTRÔLER EN TEMPS RÉEL":

III.4.2.1.1.IDENTIFICATION DES CATÉGORIES PRINCIPALES ET ATTRIBUTS :

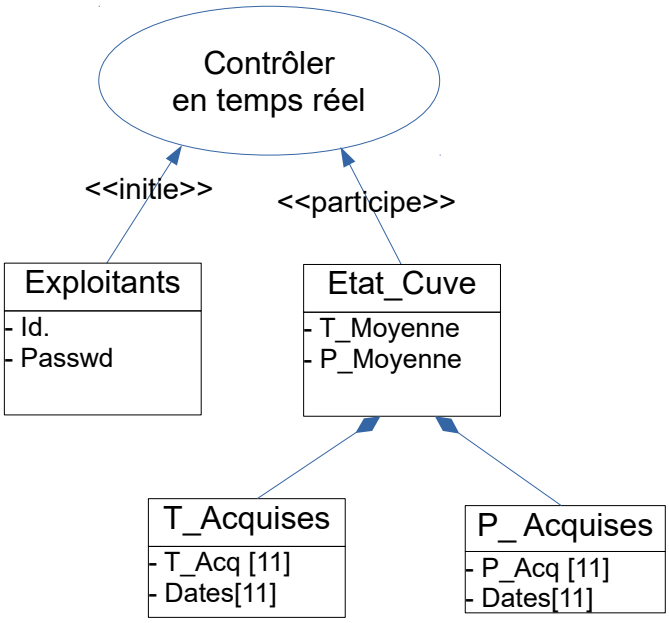
L'exigence est exprimée comme suit :

Le **PERSONNEL D'EXPLOITATION** doit pouvoir **CONTRÔLER EN TEMPS RÉEL** l'**ÉTAT COURANT DE LA CUVE** sur l'écran de son IHM.

Cet énoncé permet d'identifier les catégories suivantes :

- Le Personnel EXPLOITANT;
- L'ÉTAT COURANT DE LA CUVE.

Le diagramme de collaboration attaché à ce cas d'utilisation se présente comme suit :



NOM D'ATTRIBUT	DESCRIPTION	TYPE
Catégorie Exploitants		
Id	Identificateur de l'exploitant	string
Passwd	Mot de passe de l'exploitant	string
Catégorie État_Cuve		
T_Moyenne	Moyenne des 11 dernières températures acquises	float (degrés C)
P_Moyenne	Moyenne des 11 dernières températures	float (degrés C)

	acquises	

NOM D'ATTRIBUT	DESCRIPTION	TYPE
Catégorie T_Acquises		
T_Acq	Liste des 11 dernières pressions acquises	Float[11] (degrés C)
Dates_Acq	Liste des dates d'acq. des 11 dernières températures acquises.	float[11] (secondes)
Catégorie P_Acquises		
P_Acq	Liste des 11 dernières pressions acquises	float[11] (hectopascals)
Dates_Acq	Liste des dates d'acq. des 11 dernières pressions acquises.	float[11] (secondes)

III.4.2.1.2.IDENTIFICATION DES MÉTHODES :

Chacune des catégories modélise un objet intervenant dans le contrôle des "données cuves". Pour assimiler ces catégories à des classes, il faut identifier les méthodes qui peuvent leur être rattachées. Pour cela, pour chacune d'entre elles, il convient de se poser la question: "comment peut-on l'utiliser dans le cadre de ce cas d'utilisation" :

NOM	FONCTION	PARAMÈTRES	VALEUR RETOURNÉE
Classe Exploitant			
Connexion	Connecter l'utilisateur en tant qu'exploitant	- Id. exploitant - Mot de passe exploitant	True/false (connecté/non connecté)
Déconnexion	Se déconnecter	aucun	True/false (connecté/non connecté)
Afficher_TPD	Afficher température, pression et datation	aucun	aucune
Saisir_TC	Saisir la nouvelle température de consigne	Nouvelle température de consigne	aucun
Saisir_PC	Saisir la nouvelle pression de consigne	Nouvelle pression de consigne	aucun
Catégorie ÉTAT_CUVE			
Lire_T_Moyenne	Retourne la température moyenne	aucun	Température moyenne
Lire_P_Moyenne	Retourne la pression moyenne	aucun	Pression moyenne
Classe T_Acquises			
Ecrire_T_Acq	Écrire la température acquise dans la liste.	Température acquise	Température acquise

NOM	FONCTION	PARAMÈTRES	VALEUR RETOURNÉE
Lire_Liste_T_Acq	Extraire la liste des 11 dernières températures acquises	aucun	Liste des 11 dernières températures acquises.
Ecrire_Date_Acq	Écrire la date d'acq. de la température acquise dans la liste	Date	aucun
Lire_Liste_Dates_Acq	Extraire la liste des 11 dernières dates d'acquisition	aucun	Liste des 11 dernières dates d'acquisition.
Calculer_Moyenne_T	Calculer la moyenne des 11 dernières températures	aucun	Moyenne des 11 dernières températures acquises
Classe P_Acquises			
Ecrire_P_Acq	Écrire la pression acquise dans la liste	Pression acquise	aucun
Lire_Liste_P_Acq	Extraire la liste des 11 dernières pressions acquises	aucun	Liste des 11 dernières pressions acquises.
Lire_Dates_Acq	Extraire la liste des 11 dernières dates d'acquisition	aucun	Liste des 11 dernières dates d'acquisition.
Ecrire_Date_Acq	Écrire la date d'acq. de la pression acquise dans la liste	Date	Aucune
Calculer_Moyenne_P	Calculer la moyenne des 11 dernières pressions	aucune	Moyenne des 11 dernières pressions acquises

III.4.2.2.CAS D'UTILISATION "DÉFINIR LES VALEURS DE CONSIGNE":

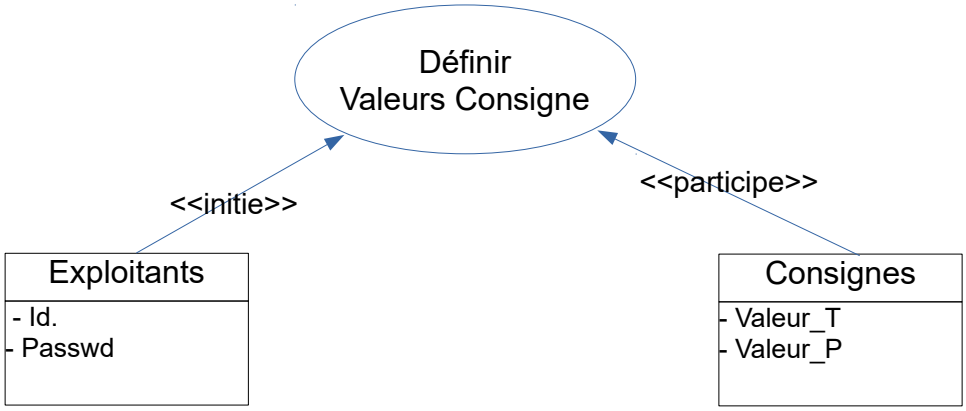
III.4.2.2.1.IDENTIFICATION DES CATÉGORIES ET DE LEURS ATTRIBUTS

L'exigence est exprimée comme suit :
Le **PERSONNEL D'EXPLOITATION** doit pouvoir définir les **VALEURS DE CONSIGNE** de la température et de la pression de la cuve.

Cet énoncé permet d'identifier les catégories suivantes :

- Le Personnel d'exploitation;
- Les valeurs de consigne

Le diagramme de collaboration attaché à ce cas d'utilisation se présente comme suit :



NOM D'ATTRIBUT	DESCRIPTION	TYPE
Catégorie Exploitants		
Id	Identificateur de l'exploitant	string
Passwd	Mot de passe de l'exploitant	string
Catégorie Consignes		
Valeur_T	Valeur de consigne pour la température	Float (degrés C)
Valeur_P	Valeur de consigne pour la pression.	Float (degrés G)

III.4.2.2.1.IDENTIFICATION DES MÉTHODES :

NOM	FONCTION	PARAMÈTRES	VALEUR RETOURNÉE
Catégorie Exploitant			
Connexion	Connecter l'utilisateur en tant qu'exploitant	- Id. exploitant - Mot de passe exploitant	True/false (connecté/non connecté)
Déconnexion	Se déconnecter	aucun	True/false (connecté/non connecté)
Catégorie Consignes			
Écrire_Consigne_T	Écrire la valeur de consigne de température	aucun	aucune
Écrire_Consigne_P	Écrire la valeur de consigne de pression	aucun	aucune

III.4.2.3.CAS D'UTILISATION "ARCHIVER":

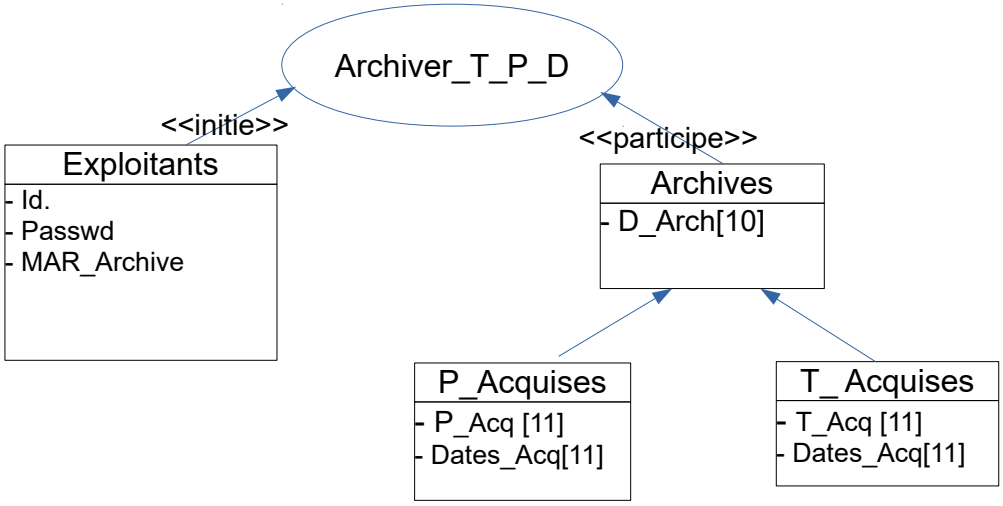
III.4.2.3.1.IDENTIFICATION DES CATÉGORIES ET DE LEURS ATTRIBUTS

L'exigence est exprimée comme suit :
Le **PERSONNEL D'EXPLOITATION** doit pouvoir **ARCHIVER** sur un support permanent les **TEMPÉRATURES** et **PRESSIONS** acquises ainsi que leurs **DATES D'ACQUISITION**.

Cet énoncé permet d'identifier les catégories suivantes :

- Le Personnel d'exploitation;
- Les températures acquises (T_Acquises) ;
- Les pressions acquises (P_Acquises) ;
- Les données archivées (Archives).

Le diagramme de collaboration attaché à ce cas d'utilisation se présente comme suit :



NOM D'ATTRIBUT	DESCRIPTION	TYPE
Catégorie Exploitants		
Id	Identificateur de l'exploitant	string
Passwd	Mot de passe de l'exploitant	string
Catégorie T_Acquises		
T_Acq	Liste des 11 dernières températures acquises	float[11] (degrés C)
Dates_Acq	Liste des dates d'acq. des 11 dernières températures acquises.	float[11] (secondes)
Catégorie P_Acquises		
P_Acq	Liste des 11 dernières pressions acquises	float[11] (hectopascals)
Dates_Acq	Liste des dates d'acq. des 11 dernières pressions acquises.	float[11] (secondes)

Catégorie Archives		
D_Archivées.D_Acq[10]	Liste des 10 dernières valeurs du trio : P_acquise, T_Acquise, Date_Acq	Tableau : float[10][3]

III.4.2.3.2.IDENTIFICATION DES MÉTHODES :

NOM	FONCTION	PARAMÈTRES	VALEUR RETOURNÉE
Catégorie Exploitant			
Connexion	Connecter l'utilisateur en tant qu'exploitant	- Id. exploitant - Mot de passe exploitant	True/false (connecté/non connecté)
Déconnexion	Se déconnecter	aucun	True/false (connecté/non connecté)
MAR_Archive	Marche ou arrêt de l'archivage	aucun	True/false (marche ou arrêt)
Catégorie T_Acquise			
Lire_V_Acq	Extraire la liste des 11 dernières températures acquises	aucun	Liste des 11 dernières températures acquises.
Lire_Dates_Acq	Extraire la liste des 11 dernières dates d'acquisition	aucun	Liste des 11 dernières dates d'acquisition.
Catégorie P_Acquise			
Lire_V_Acq	Extraire la liste des 11 dernières pressions acquises	aucun	Liste des 11 dernières pressions acquises.
Lire_Dates_Acq	Extraire la liste des 11 dernières dates d'acquisition	aucun	Liste des 11 dernières dates d'acquisition.
Catégorie Archives			
Écrire_Archive	Écrire un bloc des 10 données à archiver	- P_Acquises ; - T_Acquises ; - Dates_Acq (T) - Dates_Acq (P)	Liste des 10 dernières valeurs du trio : P_acquise, T_Acquise, Date_Acq

III.4.2.4.CAS D'UTILISATION "RÉGULER LA TEMPÉRATURE DE LA CUVE" :

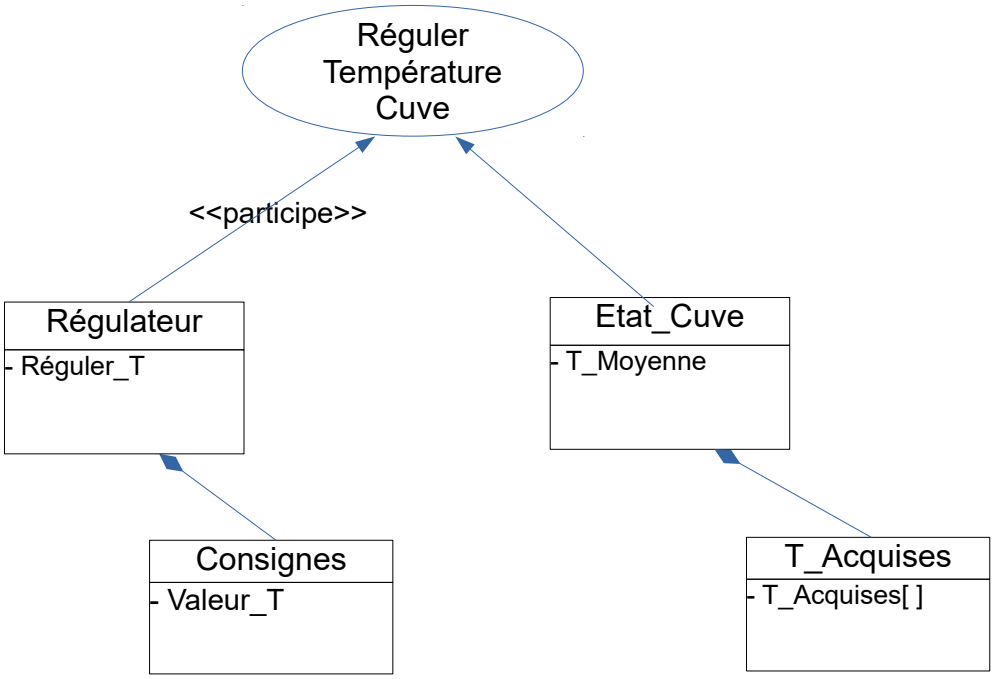
III.4.2.4.1.IDENTIFICATION DES CATÉGORIES ET DE LEURS ATTRIBUTS

L'exigence est exprimée comme suit :
L'**APPLICATION** doit **MAINTENIR AUTOMATIQUEMENT** la **TEMPÉRATURE DE LA CUVE** au voisinage de la **TEMPÉRATURE DE CONSIGNE**.

Cet énoncé permet d'identifier les catégories suivantes :

- Le Système de régulation de la température;
- La Température moyenne de la cuve (T_Moy_Cuve) ;
- La valeur de consigne de la température(Consignes_T).

Le diagramme de collaboration attaché à ce cas d'utilisation se présente comme suit :

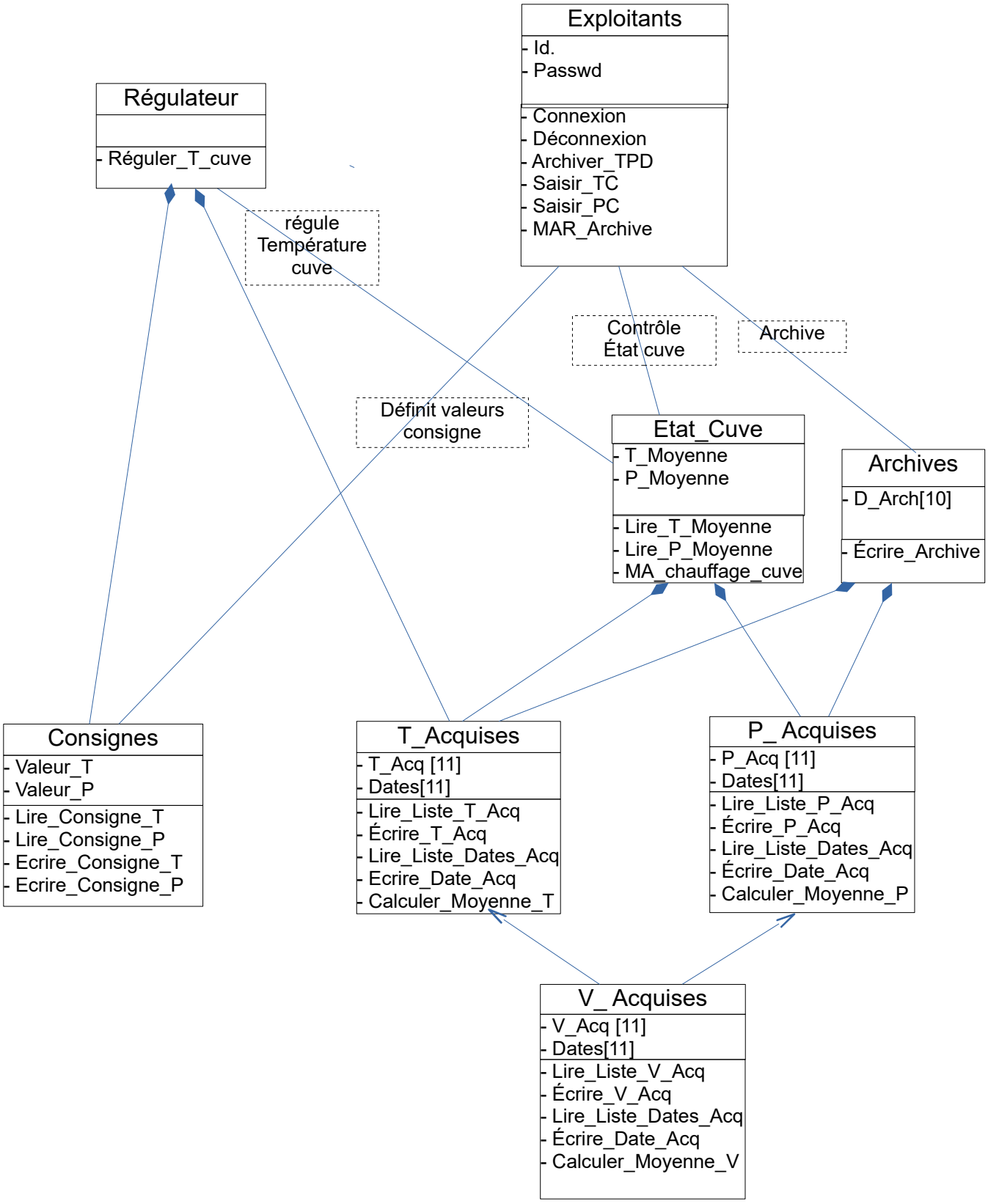


NOM D'ATTRIBUT	DESCRIPTION	TYPE
	Catégorie Consignes	
Valeur_T	Valeur de consigne pour la température	Float (degrés celsius)
Catégorie T_Acquises		
T_Acquises	11 dernières températures acquises	Float (degrés celsius)

III.4.2.4.2.IDENTIFICATION DES MÉTHODES :

NOM	FONCTION	PARAMÈTRES	VALEUR RETOURNÉE
Catégorie Consignes			
Lire_Consigne_T	Lire la valeur de la consigne de température	aucun	Valeur de consigne de température
Catégorie T_Acquise			
Lire_T_Acquise	Lire la dernière valeur de température acquise	aucun	Valeur de la dernière température acquise
Catégorie État_Cuve			
Lire_T_Moyenne		Commande M/A régulation	aucune
MA_chauffage_cuve	Commande marche/arrêt chauffage de la cuve	Commande marche/arrêt	aucune
Catégorie Régulateur			
Réguler_T_cuve	Régulation de la température de la cuve	aucun	aucune

III.4.2.5.CONSTRUCTION DU DIAGRAMME DE CLASSES DE L'APPLICATION :



III.5.SYNTHÈSE :

III.5.1.REGROUPEMENT DES ACTIVITÉS EN PROCESSUS :

III.5.1.1.SOLUTION CHOISIE POUR LE TIER "SERVEUR D'APPLICATIONS" :

III.5.1.1.1.PRINCIPES GÉNÉRAUX :

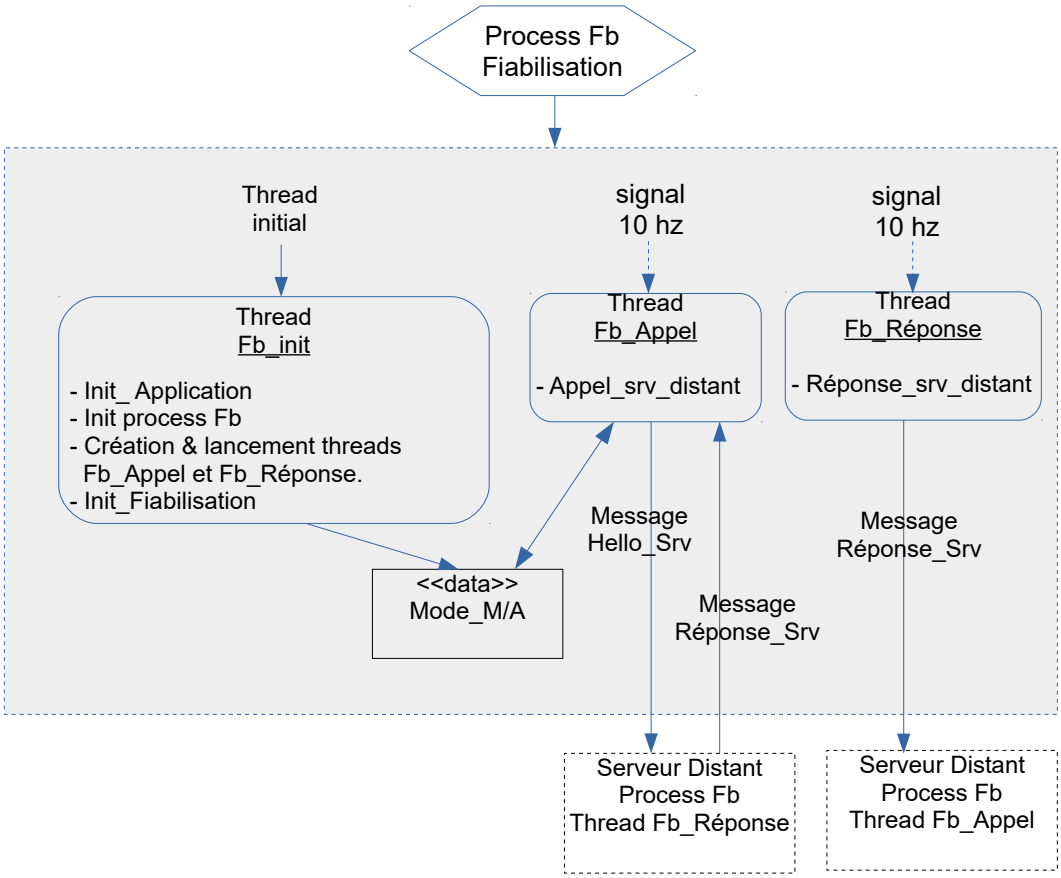
Regroupement en 4 processus :

- Process. Fb (fiabilisation) ;
- Process. Rg (régulation) ;
- Process. Ar (Archivage)
- Process Rq. (Traitement des requêtes IHM.

Suivant les critères suivants :

III.5.1.1.2.PROCESSUS Fb :

III.5.1.1.2.1.MODÉLISATION GRAPHIQUE :



III.5.1.1.2.2.COMMENTAIRES :

JUSTIFICATION DU REGROUPEMENT :

Encapsulation des traitements liés à la fiabilisation du système :
- Surveillance mutuelle des machines, détection des pannes ;
- Déclenchement du basculement maître → adjointe.

dans un seul processus qui leur est dédié, afin de limiter les effets de bords avec les activités des autres processus.

DESCRIPTION DÉTAILLEE :

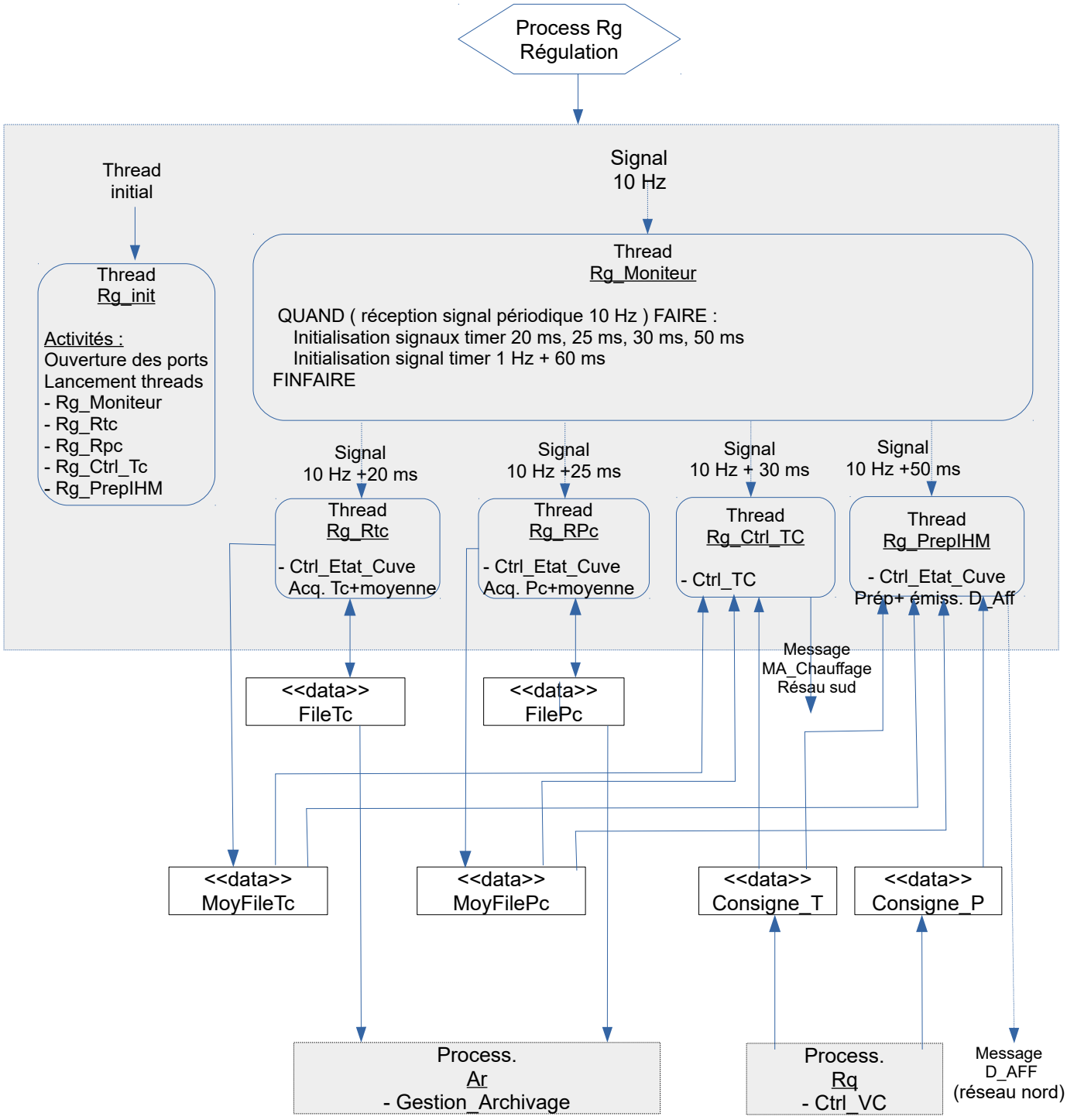
THREAD Fb_init		
CRÉATION :	Thread principal du processus Fb	
LANCEMENT INITIAL :	Au lancement du processus Fb	
DÉCLENCHEMENT :	Uniquement au lancement du processus Fb	
PRIORITÉ :	Priorité du process Fb	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Init_Application	Déclenchement du signal périodique 10 Hz
	Init_process_Fb	- Ouverture ports de communication réseau ; - Lancement threads Fb_Appel et Fb_Réponse ;
	Init_Fiabilisation	Initialisation indicateur Maître/Adjoint. FIN THREAD

THREAD Fb_Appel		
CRÉATION :	Par thread Fb_init	
LANCEMENT INITIAL :	Par thread Fb_init	
DÉCLENCHEMENT :	Cyclique (attente signal périodique 10 Hz)	
PRIORITÉ :	Priorité du process	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Appel_srv_distant	Emission message Hello_Srv au serveur distant Tempo 40 ms Lecture Réponse_Srv SI non ok et Mode_MA = adjointe ALORS Basculement en mode maître FINSI Retour attente signal 10Hz

THREAD Fb_Réponse		
CRÉATION :	Par thread Fb_init	
LANCEMENT INITIAL :	Par thread Fb_init	
DÉCLENCHEMENT :	Cyclique (attente signal périodique 10 Hz)	
PRIORITÉ :	Priorité du process	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Réponse_srv_distant	Temporisation 20 ms SI réception message Hello_Srv ALORS envoi message Réponse_Srv au serveur distant FINSI Retour attente signal 10 Hz.

III.5.1.1.3.PROCESSUS Rg :

III.5.1.1.3.1.MODÉLISATION GRAPHIQUE :



III.5.1.1.3.2.COMMENTAIRES :

JUSTIFICATION DU REGROUPEMENT :

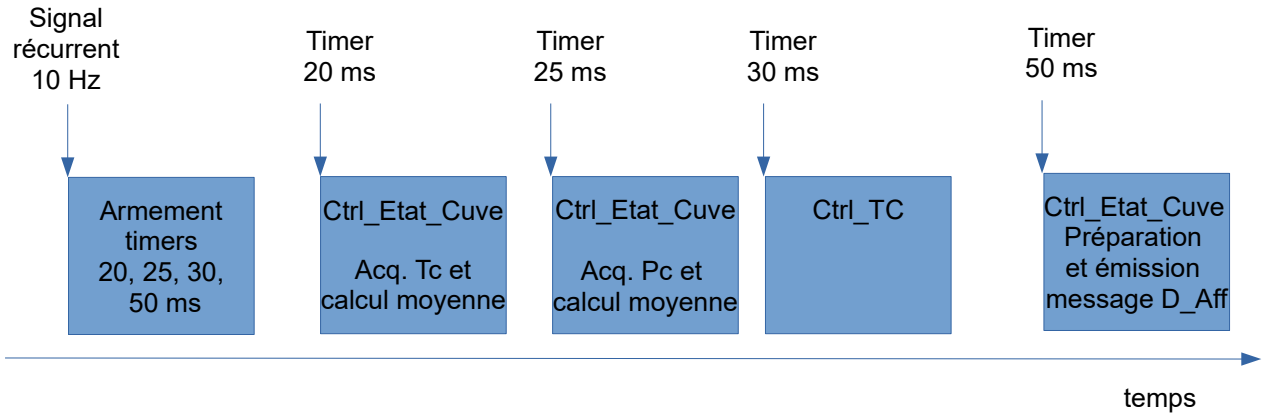
L'activité Ctrl_Etat_Cuve est récurrente et fait partie de chaînes de traitement contraintes du point de vue de la durée d'exécution.

- Ctrl_Etat_Cuve (Acquisition Tc et moyenne) → Régul_T_Cuve ;
- Ctrl_Etat_Cuve (Acquisition Tc et Pc et moyennes) → IHM (affichage).

(La flèche → matérialise les relations de précédence dans l'exécution des activités).

D'autre part, ces différentes activités sont concurrentes pour l'accès aux ressources partagées FileTc, FilePc, MoyFileTc et MoyFilePc.

L'adoption d'une programmation de type "concurrent" exigerait d'utiliser un nombre conséquent de sémaphores pour résoudre ces problèmes de précédence des traitements et d'accès aux ressources. De ce fait, nous adopterons ici une solution "synchrone" qui se matérialisera par l'inclusion d'un moniteur de tâches (thread Rg_Moniteur) chargé de déclencher les différents traitements suivant un ordonnancement temporel bien précis, matérialisé par le schéma ci-dessous.



REMARQUE : *l'activité Ctrl_Etat_Cuve se trouve ici scindée en 3 "sous-activités" qui seront déclenchées d'une manière synchrone pour éviter d'avoir à traiter les conflits d'accès aux ressources communes.*

Les avantages de cette méthode sont :

- Elle garantit le respect des relations de précédence entre activités ;
- Elle évite toute concurrence sur les ressources communes.

REMARQUE : *L'activité Archivage est, elle aussi, récurrente (récurrence 1 Hz) et liée à*

l'activité Ctrl_Etat_Cuve par une relation de précédence et elle utilise les ressources FileTc et FilePc. Cependant, la solution de l'encapsuler dans le processus Rg n'a pas été retenue pour éviter qu'un dysfonctionnement du système physique d'archivage ne perturbe le fonctionnement de la régulation de la température de la cuve. Cependant, pour éviter tout conflit avec les activités du processus Rg et respecter la précédence de l'activité Ctrl_Etat_Cuve avec Archivage, cette dernière activité sera déclenchée 60 ms après la survenue du signal 1 Hz.

DESCRIPTION DÉTAILLEE :

THREAD Rg_init		
CRÉATION :	Thread principal du processus Rg	
LANCEMENT INITIAL :	Au lancement du processus Rg	
DÉCLENCHEMENT :	Uniquement au lancement du processus Rg	
PRIORITÉ :	Priorité du process Rg	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Init_Rg	Ouverture des ports réseau : - Réception Tc ; - Réception Pc ; - Emission message M/A chauffage ; - Emission message D_Aff ; Lancement threads : - Rg_Moniteur - Rg_Rtc - Rg_Rpc - Rg_Regul_T_cuve - Rg_PrepIHM.

THREAD Rg_Moniteur		
CRÉATION :	Thread principal du processus Rg	
LANCEMENT INITIAL :	Au lancement du processus Rg	
DÉCLENCHEMENT :	Signal 10 Hz	
PRIORITÉ :	Priorité du process Rg	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
		Initialisation des timers de déclenchement des threads : - Rg_Rtc - Rg_Rpc - Rg_Regul_T_cuve - Rg_PrepIHM.

THREAD Rg_RTc	
CRÉATION :	Par thread Ini_Rg
LANCEMENT INITIAL :	Par thread Ini_Rg

DÉCLENCHEMENT :	Périodique sur réception du signal de temporisation 10 Hz + 20 ms	
PRIORITÉ :	Priorité du process Rg	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Acq_T_Cuve	Quand (réception signal tempo 20 ms) Lecture message Tc dans le buffer de réception réseau ; Introduction valeur reçue dans la file FileTC ; Calcul de la moyenne glissante des 11 dernières Tc et sauvegarde dans MoyFileTc ; Retour en attente réception signal tempo 20 ms ;

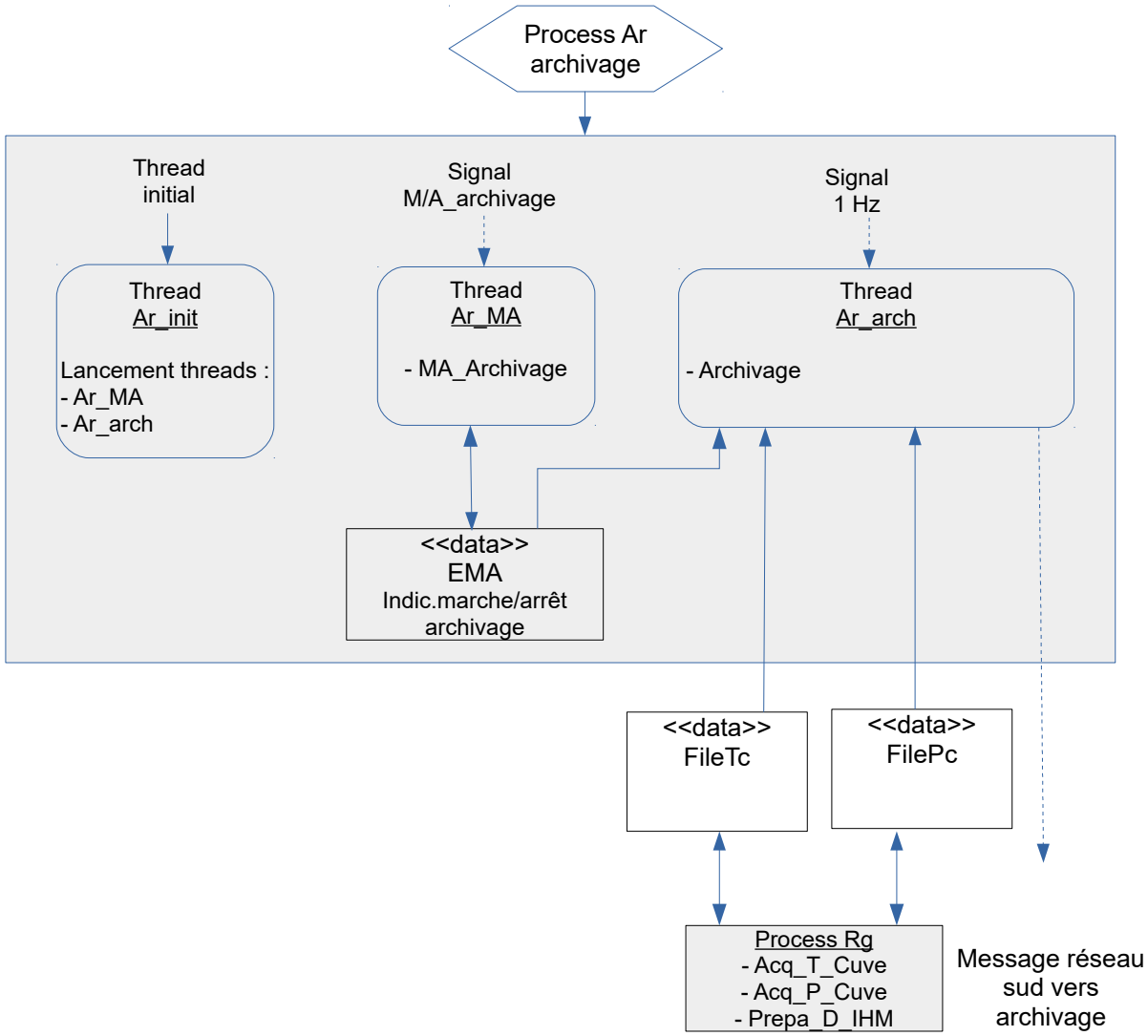
THREAD Rg_RPc		
CRÉATION :	Par thread Ini_Rg	
LANCEMENT INITIAL :	Par thread Ini_Rg	
DÉCLENCHEMENT :	Périodique sur réception du signal de temporisation 10 Hz + 25 ms	
PRIORITÉ :	Priorité du process Rg	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Acq_T_Cuve	Quand (réception signal tempo 20 ms) Lecture message Tc dans le buffer de réception réseau ; Introduction valeur reçue dans la file FileTC ; Calcul de la moyenne glissante des 11 dernières Tc et sauvegarde dans MoyFileTc ; Retour en attente réception signal tempo 20 ms ;

THREAD Rg_Regul_T		
CRÉATION :	Par thread Ini_Rg	
LANCEMENT INITIAL :	Par thread Ini_Rg	
DÉCLENCHEMENT :	Périodique sur réception du signal de temporisation 10 Hz + 30 ms	
PRIORITÉ :	Priorité du process Rg	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Régul_T_Cuve	Régulation de la température cuve autour de la température de consigne.

THREAD Rg_PrepIHM		
CRÉATION :	Par thread Ini_Rg	
LANCEMENT INITIAL :	Par thread Ini_Rg	
DÉCLENCHEMENT :	Périodique sur réception du signal de temporisation 10 Hz + 50 ms	
PRIORITÉ :	Priorité du process Rg	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
		Préparation du message D_AFF destiné à l'élaboration des vues à afficher sur l'IHM et émission de ce message sur le réseau nord à destination du processus d'affichage de l'IHM

III.5.1.1.4.PROCESSUS Ar:

III.5.1.1.4.1.MODÉLISATION GRAPHIQUE :



III.5.1.1.4.2.COMMENTAIRES :

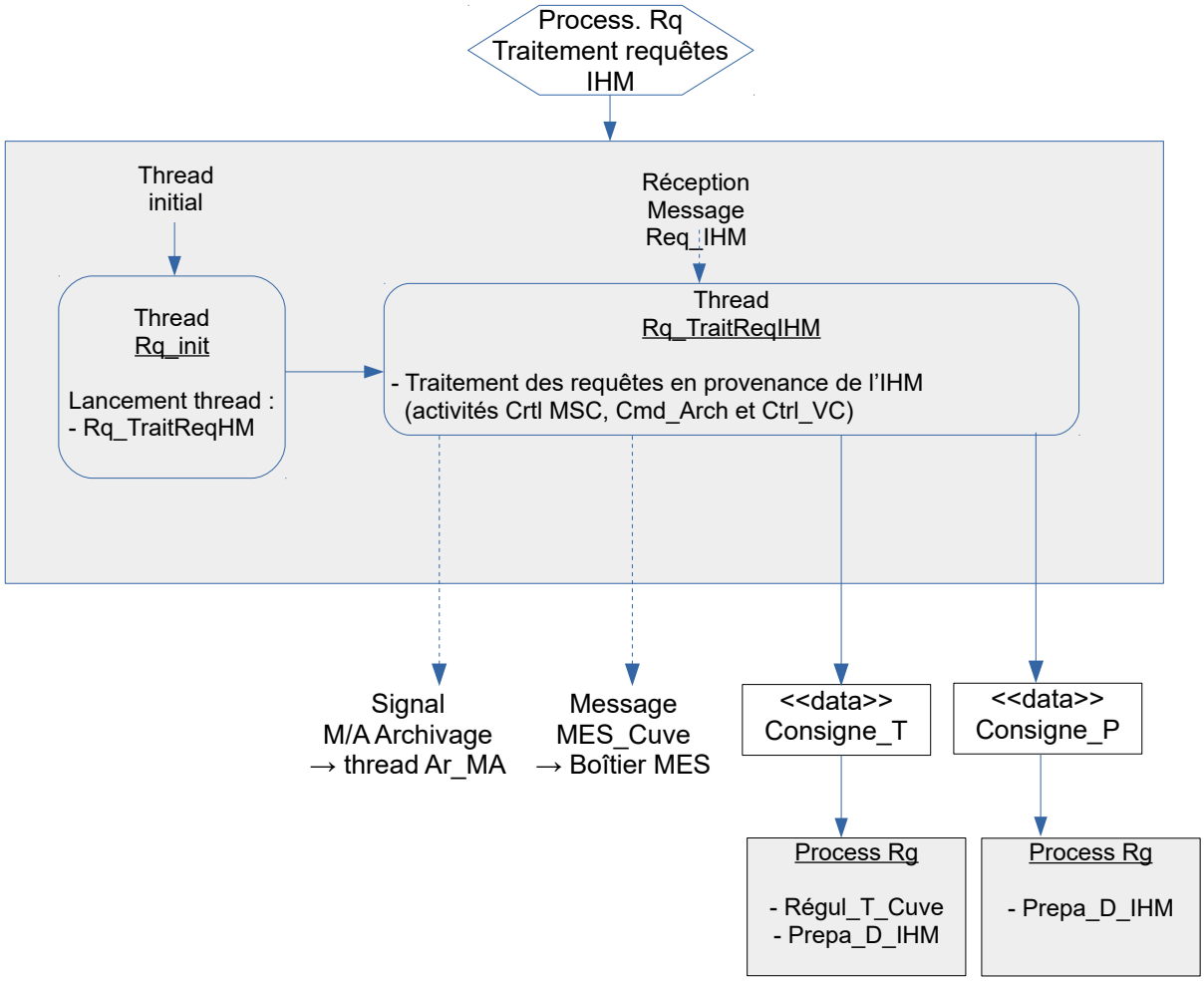
THREAD Ar_init		
CRÉATION :	Thread initial du processus Ar	
LANCEMENT INITIAL :	Lancement initial du processus Ar	
DÉCLENCHEMENT :	Unique	
PRIORITÉ :	Priorité du process Ar	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Init_Archivage	- Init. Indicateur EMA à la valeur "arrêt" ; - Lancement threads Ar_MA et Ar_arch.

THREAD Ar_MA		
CRÉATION :	Par thread Ar_init (à la création de celui-ci)	
LANCEMENT INITIAL :	Par thread Ar_init	
DÉCLENCHEMENT :	Aléatoire, à chaque réception du signal M/A_Archivage	
PRIORITÉ :	Priorité du process Ar	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	MA_Archivage	Quand (réception signal M/A_Archivage) FAIRE SI (EMA = "arrèt") ALORS EMA = "marche" ; SINON (EMA = "arrêt") Retour attente réception signal M/A_Archivage

THREAD Ar_arch		
CRÉATION :	Par thread Ar_init (à la création de celui-ci)	
LANCEMENT INITIAL :	Par thread Ar_init	
DÉCLENCHEMENT :	Périodique, à chaque réception du signal 1 hz + 60 ms	
PRIORITÉ :	Priorité du process Ar	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Archivage	QUAND (réception signal 1 hz + 60 ms) FAIRE Préparation du bloc d'enregistrement ; Lancement de l'écriture du bloc dans le système d'archivage ; Retour attente signal 1 hz + 60 m.

III.5.1.1.5.PROCESSUS Rq :

III.5.1.1.5.1.MODÉLISATION GRAPHIQUE :



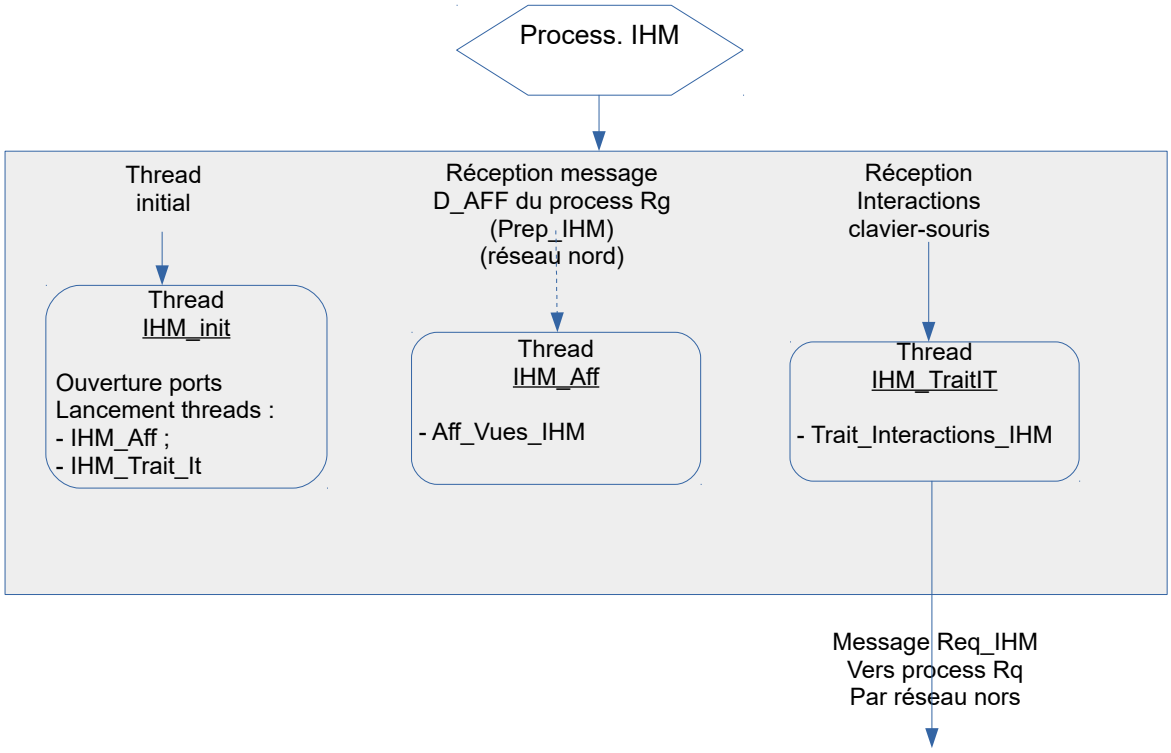
III.5.1.1.5.2.COMMENTAIRES :

THREAD Rq_init		
CRÉATION :	Thread initial du processus Ar	
LANCEMENT INITIAL :	Lancement initial du processus Rq	
DÉCLENCHEMENT :	Unique	
PRIORITÉ :	Priorité du process Ar	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
		- Lancement thread Rq_TraitReqIHM

THREAD Rq_TraitReqIHM		
CRÉATION :	Par thread Rq_Init (à la création de celui-ci)	
LANCEMENT INITIAL :	Par thread Rq_Init	
DÉCLENCHEMENT :	Aléatoire : a chaque réception d'une requête IHM en provenance du réseau nord (IHM)	
PRIORITÉ :	Priorité du process Rq	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	TraitReqIHM	QUAND (réception message Req_IHM) FAIRE SUIVANT LE CAS (type de requête transmise) FAIRE - MES_Cuve : émission message mise en sécurité cuve vers boîtier mise en sécurité ; - M/A Archivage : Emission signal M/A_Archivage vers process Ar ; - Saisie consigne T : sauvegarde valeur saisie dans Consigne_T ; - Saisie consigne P : sauvegarde valeur saisie dans Consigne_P. FINCAS Retour attente réception message Req_IHM.

III.5.1.1.6.PROCESSUS IHM:

III.5.1.1.6.1.MODÉLISATION GRAPHIQUE :



III.5.1.1.6.2.COMMENTAIRES :

THREAD IHM_init		
CRÉATION :	Thread initial du process. IHM	
LANCEMENT INITIAL :	Lancement process IHM	
DÉCLENCHEMENT :	Unique	
PRIORITÉ :	Priorité du process IHM	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Init_IHM	Initialisation du processus IHM, ouverture des ports, lancement des threads de traitement.

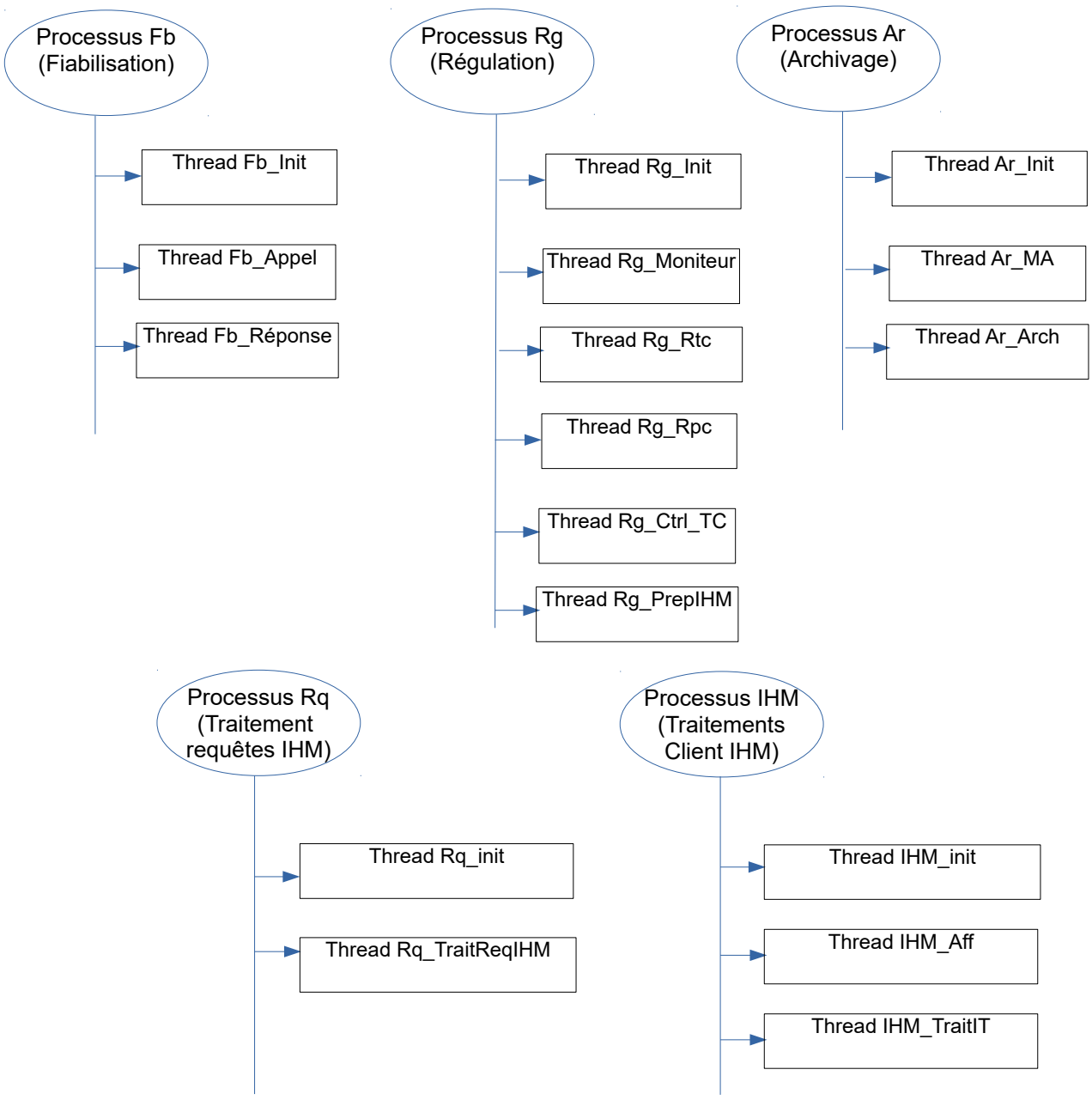
THREAD IHM_Aff		
CRÉATION :	Par thread Init_IHM	
LANCEMENT INITIAL :	Par thread Init_IHM	
DÉCLENCHEMENT :	Périodique (réception message D_AFF sur réseau nord)	
PRIORITÉ :	Priorité du process IHM	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Aff_Vues_IHM	Rafraîchissement des vues de l'IHM à partir des données du message D_AFF reçu.

THREAD IHM_TraitIT		
CRÉATION :	Par thread Init_IHM	
LANCEMENT INITIAL :	Par thread Init_IHM	
DÉCLENCHEMENT :	Aléatoire (par événement interaction clavier-souris)	
PRIORITÉ :	Priorité du process IHM	
TRAITEMENTS SUPPORTÉS :	**Nom**	**Traitements**
	Trait_Interactions_IHM	Acquisition et traitement des interactions opérateurs en provenance de l'IHM.

IV.CONCLUSION :

A l'issue de cette étude, nous disposons :

D'une architecture "dynamique" du logiciel décrivant la répartition des traitements en différents processus et threads. Cette architecture peut être représentée de la manière suivante :



Les niveaux de priorité des processus ainsi que les traitements supportés par les différents threads et leurs modalités de déclenchement sont également décrits. Les liens

entre ces traitements et les activités de l'application sont établis.

D'autre part, nous disposons d'une architecture "statique" issue de l'étude du point de vue logique.

Enfin, nous disposons d'une architecture "système" qui permet d'identifier les différentes évolutions à réaliser sur le système informatique d'accueil.

A partir de ces acquis, le responsable du projet est à même d'élaborer le PLAN DE DÉVELOPPEMENT du projet : tâches à effectuer, répartition de ces tâches entre les équipes, planification du développement, ce qui lui permettra d'organiser le travail des équipes en vue de la conception détaillé, de la réalisation et des tests unitaires des différents modules de réalisation.

En particulier, la conception détaillée et la réalisation du logiciel pourra être fractionnée d'une manière rationnelle en différents modules pouvant être réalisés en parallèle par les développeurs ou les équipes de développement (en interne ou en sous-traitance).