

DBA CHEZ FIVES SYLEPS
SIMULATION ENTRETIEN
ORACLE

BY IDEO-LAB

10TH FEB 2026



Le jeu des questions/réponse

1 "Pourquoi voulez-vous rejoindre Fives Syleps ?"

Réponse prête à dire :

Ce qui m'intéresse particulièrement chez Fives Syleps, c'est le lien direct entre les systèmes d'information et le monde industriel réel. Ici, la base de données ne supporte pas seulement une application métier classique : elle soutient des flux logistiques et industriels en production, avec un impact immédiat sur le terrain.

J'apprécie beaucoup cet environnement où la performance et la disponibilité des bases ont des conséquences concrètes sur l'exploitation des sites. Mon profil de DBA orienté continuité de service, performance sous charge et fiabilité des données correspond parfaitement à ce type de contexte critique.

 Fives Syleps



2 “Comment voyez-vous le rôle d’un DBA chez nous ?”

Chez Fives Syleps, le DBA a selon moi un rôle bien plus large qu’un simple administrateur technique. Il est un acteur clé de la continuité des opérations logistiques et industrielles.

La base de données est au cœur des WMS, WCS et outils de supervision qui pilotent les flux physiques. Une indisponibilité ou une dégradation de performance peut bloquer des flux logistiques ou de production. Le DBA devient donc un garant de la disponibilité, de la performance et de la cohérence des données, avec une forte dimension opérationnelle et un lien direct avec le métier.



Fives Syleps

3 “Parlez-nous de votre approche sur la disponibilité des bases”

Dans un environnement comme le vôtre, où les systèmes peuvent fonctionner sur des plages horaires étendues voire en 24/7, la disponibilité est prioritaire.

Mon approche repose sur trois axes :

- la prévention, via une supervision fine, l'anticipation de la saturation (CPU, I/O, stockage) et des tests réguliers de sauvegarde/restauration ;
- la résilience, avec des architectures de réplication ou de PRA adaptées aux contraintes RPO/RTO ;
- la réactivité, avec des procédures d'incident claires pour réduire au maximum les temps d'arrêt.

L'objectif est que la base de données ne soit jamais un point de blocage des flux

4 “Comment gérez-vous la performance dans des systèmes transactionnels ?”

Les environnements logistiques génèrent beaucoup de transactions, des mises à jour fréquentes et parfois des traitements batch massifs. Dans ce contexte, je travaille à plusieurs niveaux :

- optimisation SQL et analyse des plans d'exécution,
- mise en place d'index pertinents et revue régulière de leur efficacité,
- surveillance des verrous et contentions,
- suivi des statistiques et de la volumétrie pour anticiper les dérives.

Le but est de maintenir des temps de réponse compatibles avec des processus industriels en temps réel ou quasi temps réel ce qui est clairement un enjeu fort chez

5 “Quelle importance accordez-vous à l’intégrité et à la sécurité des données ?”

Dans un contexte industriel et logistique, les données représentent la traçabilité des produits, l’état des systèmes et l’historique des opérations. Leur intégrité est essentielle.

Je veille à la cohérence des données via des sauvegardes fiables, des contrôles réguliers et une gestion stricte des accès. Côté sécurité, je m’assure que les droits sont correctement segmentés, que les accès sont tracés et que l’audit est possible.

Pour moi, la base doit être à la fois performante, mais aussi maîtrisée et sécurisée.

 Fives Syleps

6 “Comment vous situez-vous entre RUN et projets ?”

Je me vois comme un DBA à la fois opérationnel et structurant.

Côté RUN, je suis très attaché à la supervision proactive, au traitement rapide des incidents et à l'analyse des blocages ou lenteurs, car cela impacte directement la production.

Mais j'accorde aussi beaucoup d'importance à la partie administration et optimisation : configuration des instances, gestion des droits, optimisation des index et accompagnement des équipes projet pour que les bases soient bien dimensionnées et performantes dès le départ. C'est cette combinaison RUN + amélioration continue qui apporte de la stabilité sur le long terme.

 Fives Syleps


7 “Comment travaillez-vous avec les autres équipes ?”

Le DBA est par nature transversal. Dans un contexte comme Fives Syleps, je travaillerais étroitement avec :

- les développeurs, pour les aider sur les requêtes critiques et les modèles de données,
- les équipes systèmes et réseaux, pour tout ce qui touche à l'infrastructure et à la performance globale,
- les équipes projet, lors des déploiements sur les sites clients,
- et parfois les équipes d'exploitation côté client.

J'essaie toujours de traduire les contraintes métier – ici les flux logistiques et industriels – en exigences techniques sur la base de données.

 Fives Syleps



DISPONIBILITÉ
24/7 —
Continuité de
service

Dans un contexte comme le vôtre, la base ne doit jamais être un point de défaillance des flux logistiques.

Je travaille donc sur trois niveaux :

1. Prévention

Supervision proactive des indicateurs clés : tablespaces, I/O, CPU, sessions, verrous, latence disque. L'objectif est d'anticiper les saturations avant impact.

2. Résilience

Mise en place de stratégies de reprise : sauvegardes RMAN avec tests réguliers de restauration, archivage des redo logs sécurisé, et si le niveau de criticité l'exige, Data Guard pour bascule rapide.

3. Réactivité

? ***“Que faites-vous pour être prêt à un crash base en pleine proc***

Je considère qu'un PRA n'existe que s'il est testé.

- ✓ Sauvegardes automatisées et surveillées
- ✓ Tests de restauration réguliers sur environnement isolé
- ✓ Vérification des archives redo
- ✓ Documentation des procédures de recovery

L'objectif est de pouvoir restaurer dans des délais compatibles avec la con opérations logistiques.

PERFORMANCE — Temps
réel & forte volumétrie

? ***“Comment gérez-vous une base avec beaucoup de transactions et mises à jour fréquentes ?”***

Les systèmes logistiques génèrent énormément de transactions courtes, souvent liées à des changements de statut ou des événements terrain.

Je me concentre sur :

- ◆ **Réduction des contentions** : analyse des verrous (V\$LOCK), gestion des index pour éviter les hot blocks
- ◆ **Optimisation SQL** : AWR, ASH, plans d'exécution pour identifier les requêtes critiques
- ◆ **Gestion des statistiques** : stats à jour pour éviter les mauvais plans
- ◆ **Surveillance des I/O** : car dans ces systèmes, le disque est souvent le facteur limitant

? ***“Comment traitez-vous les requêtes lentes en production ?”***

Je commence par identifier si le problème est SQL, I/O ou verrouillage.

1. “AWR/ASH pour trouver les requêtes dominantes”
2. “Analyse du plan d’exécution”
3. “Vérification des index, cardinalités et stats”
4. “Si besoin, création ou ajustement d’index ou réécriture SQL”

L’objectif est toujours de réduire le temps de réponse sans perturber la production.

? "Et les batchs lourds ?"

Les batchs peuvent dégrader les performances temps réel s'ils sont mal planifiés.

Je veille à :

- ✓ Les planifier hors pics d'activité
- ✓ Surveiller leur impact I/O et undo/redo
- ✓ Optimiser les traitements de masse (bulk, parallélisme maîtrisé)



INTÉGRITÉ DES
DONNÉES — Traçabilité
critique

? ***“Comment garantissez-vous la fiabilité des données ?”***

Dans un environnement industriel, les données reflètent des mouvements physiques réels.

Je veille à :

- ✓ Sauvegardes cohérentes et vérifiées
- ✓ Monitoring des corruptions potentielles
- ✓ Surveillance des erreurs applicatives récurrentes
- ✓ Intégrité transactionnelle (gestion des undo, rollback segments)



SÉCURITÉ & AUDIT

? ***“Quelle est votre approche de la sécurité Oracle ?”***

Je travaille sur trois axes :

- 🔒 **Contrôle des accès** : rôles, privilèges minimum nécessaires
- 🔒 **Traçabilité** : audit des connexions et opérations sensibles
- 🔒 **Sécurisation des échanges** : chiffrement des connexions si nécessaire

Le but est que seules les personnes autorisées accèdent aux données critiques.



GESTION DES INCIDENTS (RUN)



? "Que faites-vous face à une base lente en pleine production ?"

Réaction structurée :

1. "Vérification charge système (CPU/I/O)"
2. "Analyse des sessions actives"
3. "Recherche de verrous bloquants"
4. "Identification des requêtes dominantes"

Je privilégie toujours une action ciblée et mesurée pour stabiliser rapidement la situation sans risque.

? ***“Comment gérez-vous les verrous bloquants ?”***

Identification via vues dynamiques (sessions bloquantes/bloquées), compréhension du SQL en cause, et action soit côté optimisation, soit en coordination avec l'équipe applicative.



TRAVAIL TRANSVERSE

? ***“Comment collaborez-vous avec les équipes applicatives ?”***

Le DBA ne travaille pas isolé.

J'aide les équipes à :

- ✓ Comprendre l'impact SQL
- ✓ Concevoir des requêtes performantes
- ✓ Anticiper les impacts volumétriques

Le but est d'éviter les incidents plutôt que de les subir.



QUESTIONS PIEGES ET
TECHNIQUES

 **1** *“Une base est lente, mais le CPU est bas. Où regardez-vous ?”*

 Ils testent ta vision I/O & verrous

Réponse :

Si le CPU est bas, je regarde immédiatement les I/O et les attentes.

- Top wait events (AWR/ASH)
- db file sequential read / scattered read
- log file sync
- file I/O latency

Ensuite je vérifie les verrous et la contention (TX, TM).

Une base peut être “lente” sans CPU saturé si elle attend le disque ou un verrou.

🔥 2 **“Pourquoi une requête rapide hier devient lente aujourd’hui ?”**

👉 Ils testent stats + plan d’exécution

Réponse :

Changement de plan d’exécution.

Causes classiques :

- Statistiques obsolètes ou rafraîchies avec distribution différente
- Volume de données fortement augmenté
- Bind peeking / adaptive cursor sharing
- Nouvel index ou suppression d’index

Je compare les plans (DBMS_XPLAN) et je vérifie l’historique AWR.

🔥 3 ***“Différence entre db file sequential read et scattered read ?”***

👉 Fondamental Oracle

Réponse :

Sequential read = lecture mono-bloc → index access

Scattered read = lecture multi-blocs → full table scan

Si je vois beaucoup de scattered read sur une table critique, je regarde les index manquants ou un plan inadapté.

4 **“Comment diagnostiquer un problème de verrous ?”**

Réponse :

Je cherche :

- Sessions bloquantes vs bloquées
- Type de verrou (TX = row lock, TM = table lock)
- SQL en cours

Puis j'identifie la cause : transaction trop longue, index manquant causant full scan + lock étendu, ou logique applicative.

5 *“Redo log saturé, que se passe-t-il ?”*

 Contexte transactionnel fort comme logistique

Réponse :

Si les redo sont trop petits ou mal dimensionnés :

- Switch trop fréquents
- Attentes log file switch / checkpoint incomplete

Impact : ralentissement global des transactions.

Solution : augmenter taille redo logs et optimiser checkpoint.

6 “*UNDO saturé, symptômes ?*”

Réponse :

ORA-01555 snapshot too old

Requêtes longues qui échouent

Solution :

- Augmenter UNDO tablespace
- Ajuster undo_retention
- Éviter transactions trop longues

7 *“Pourquoi un index peut dégrader les performances ?”*

 Question piège classique

Réponse :

Trop d'index =

- Ralentissement des INSERT/UPDATE/DELETE
- Maintenance accrue
- Mauvais plan si stats fausses

Un index doit être justifié par un gain lecture supérieur au coût écriture.

8 *“Que fait Oracle quand un disque devient lent ?”*

Réponse :

Les sessions attendent sur des wait events I/O.

Le temps DB augmente sans hausse CPU.

On le voit via AWR : temps passé sur I/O wait.

Le DBA doit corréliser avec l'équipe système.

9 ***“Différence entre Data Guard et RMAN ?”***

Réponse :

RMAN = sauvegarde/restauration

Data Guard = réplication temps réel pour PRA

RMAN protège contre perte de données

Data Guard protège contre indisponibilité du site

10 *“Que surveillez-vous en priorité sur une base critique ?”*

Réponse structurée :

1. “Espace tablespaces”
2. “I/O latency”
3. “Sessions actives / verrous”
4. “Top SQL”
5. “Redo generation rate”
6. “Jobs en échec”

L'objectif est d'anticiper avant incident.

“Pourquoi un DBA est crucial dans un environnement logistique comme le nôtre ?”

Parce qu'ici la base de données ne supporte pas juste un logiciel, elle soutient des flux physiques réels. Une dégradation DB peut bloquer la production ou la logistique. Le DBA est garant de la continuité opérationnelle.



mises en situation réelles



SCÉNARIO 1 — La base devient très lente en pleine production

Contexte

Les opérateurs se plaignent : les écrans mettent plusieurs secondes à répondre. Les flux logistiques ralentissent.

Réponse attendue

Je commence par qualifier le problème sans agir dans la précipitation.

- 1 Vérifier si c'est global ou limité à certaines sessions
- 2 Regarder la charge système : CPU, I/O, mémoire
- 3 Identifier les principales attentes (AWR/ASH)

Ensuite :

- "Si attente I/O → possible saturation disque"
- "Si verrous → identifier session bloquante"
- "Si SQL dominant → analyser plan d'exécution"

Mon objectif est de stabiliser rapidement, puis d'analyser en profondeur après coup.

SCÉNARIO 2 — Un batch nocturne bloque la production le matin

Contexte

Un traitement de masse a tourné plus longtemps que prévu et impacte les performances au démarrage des équipes.

Réponse attendue

Je vérifie d'abord si le batch est encore actif et ses consommations I/O et UNDO.

Ensuite :

- "Vérifier s'il génère des verrous bloquants"
- "Identifier son SQL principal"

Actions possibles :

- ✓ Ajuster planification (hors pics)
- ✓ Optimiser requêtes batch
- ✓ Ajouter index adaptés
- ✓ Fractionner le traitement en lots

Le but est d'éviter qu'un traitement de masse pénalise les flux temps réel.

SCÉNARIO 3 — ORA-01555 “snapshot too old” en production

Contexte

Une requête de supervision échoue, impactant la visibilité sur les flux.

Réponse attendue

Cela indique un problème UNDO.

Je vérifie :

- “Taille du UNDO tablespace”
- “undo_retention”
- “Volume de transactions longues”

Solution :

- ✓ Augmenter UNDO
- ✓ Éviter requêtes longues sur données très volatiles
- ✓ Adapter stratégie de batch

SCÉNARIO 4 — Blocage massif de sessions

Contexte

Des dizaines de sessions sont bloquées, plus aucun flux n'avance.

Réponse attendue

Je cherche immédiatement la session bloquante via les vues dynamiques.

J'analyse :

- "Type de verrou (row vs table)"
- "SQL exécuté"

Si transaction anormale :

- ✓ Contact équipe applicative
- ✓ Kill session si nécessaire (procédure encadrée)

Ensuite analyse préventive : index manquant ? transaction trop longue ?

SCÉNARIO 5 — Crash base brutal

Contexte

La base tombe. Les systèmes logistiques sont à l'arrêt.

Réponse attendue

Priorité : restauration rapide du service.

- 1** Diagnostic : instance crash ou corruption ?
- 2** Startup mount → recovery automatique
- 3** Si nécessaire, restauration RMAN

Ensuite seulement :

- “Analyse des causes (I/O ? OS ? bug ?)”
- “Rapport d’incident”
- “Actions préventives”

SCÉNARIO 6 — Un index manquant fait exploser les temps de réponse

Contexte

Une requête clé du WMS prend 10 secondes au lieu de 100 ms.

Réponse attendue

Je récupère le SQL et son plan d'exécution.

Si full table scan sur grosse table :

- ✓ Vérifier cardinalité
- ✓ Créer index adapté
- ✓ Mettre à jour statistiques

Toujours valider le gain avant mise en production.

SCÉNARIO 7 — Disque lent sur serveur DB

Contexte

Pas d'erreur Oracle, mais tout est lent.

Réponse attendue

Je regarde les wait events I/O.

Si latence élevée :

- ✓ Corrélation avec équipe système
- ✓ Vérifier filesystems / SAN

Oracle attend simplement le disque → ce n'est pas un problème SQL.

SCÉNARIO 8 — Croissance rapide des données

Contexte

Les tables de traçabilité explosent en volumétrie.

Réponse attendue

Anticipation :

- ✓ Partitionnement des tables volumineuses
- ✓ Archivage des données anciennes
- ✓ Surveillance tablespaces

Objectif : maintenir performance et maîtriser la taille.

“Dans un environnement logistique comme le vôtre, un incident base n’est pas un incident IT classique, c’est un incident opérationnel. Mon rôle est d’agir vite, de manière structurée, et surtout de mettre en place les mécanismes pour que ces incidents deviennent rares.”

Les 10 erreurs fatales qu'un DBA ne doit
jamais faire en production

1 Modifier en prod sans comprendre l'impact

Changer un paramètre, un index ou une structure sans analyse peut dégrader les performances ou bloquer l'application.

Toujours mesurer, tester, planifier un rollback.

2 Lancer une opération lourde en pleine charge

Rebuild d'index, stats globales, export massif... en heures de pointe = risque de saturation I/O et blocage des flux.

Toujours planifier hors période critique.

3 Supprimer un index "qui semble inutile"

Un index peut paraître peu utilisé mais être critique pour une requête métier rare mais essentielle.

Toujours analyser AWR/ASH avant suppression.

4 Négliger les sauvegardes... ou ne jamais tester les restaurations

Une sauvegarde non testée n'est pas une sauvegarde.

En environnement industriel, l'impossibilité de restaurer peut bloquer totalement l'activité.

5 Tuer une session sans comprendre ce qu'elle fait

Un kill session peut provoquer rollback massif, blocage prolongé, ou corruption logique côté applicatif.

Toujours identifier le SQL et la transaction.

6 Laisser les tablespaces arriver à saturation

Une base qui ne peut plus écrire = arrêt immédiat des transactions.

La surveillance proactive de l'espace disque est vitale.

7 Mettre à jour les statistiques sans stratégie

Des stats mal rafraîchies peuvent changer brutalement les plans d'exécution et dégrader les performances.

Toujours cibler et comprendre les impacts.

8 Appliquer un patch ou redémarrer sans plan de retour arrière

Même un patch mineur peut poser problème.

Toujours prévoir rollback, sauvegarde préalable, et fenêtre maîtrisée.

9 Ignorer les signaux faibles

Petites alertes répétées (I/O en hausse, UNDO qui grossit, verrous fréquents) sont souvent les prémices d'un incident majeur.

Le rôle du DBA est d'anticiper.

10 Rester isolé

Un DBA qui ne communique pas avec les équipes applicatives ou systèmes peut créer des incidents involontaires.

En environnement critique, la coordination est aussi importante que la technique.

Les qualités humaines d'un DBA senior en environnement critique

1 Sang-froid sous pression

En production critique, il y a parfois des incidents avec un impact direct sur l'activité. Le DBA doit garder la tête froide, analyser les faits, et agir de manière structurée. La panique est souvent plus dangereuse que le problème technique.

2 Sens des responsabilités

Une base de données peut bloquer une chaîne logistique ou industrielle. Le DBA doit être conscient de l'impact réel de ses actions et prendre ses décisions avec prudence et rigueur.

3 Rigueur et méthode

Diagnostic, changement, patch, optimisation... tout doit être fait avec une démarche claire : analyse → action → vérification → documentation. En environnement critique, l'improvisation est risquée.

4 Capacité à prioriser

Tous les problèmes ne sont pas urgents.

Un DBA senior doit savoir distinguer ce qui menace la production immédiatement de ce qui peut attendre, pour concentrer ses efforts au bon endroit.

5 Communication claire

Le DBA est au carrefour des équipes : dev, systèmes, réseaux, support, parfois client.

Il doit expliquer les problèmes techniques avec des mots compréhensibles, surtout en situation d'incident.

6 Esprit d'équipe

La base de données ne vit pas seule.

Les performances dépendent souvent de l'application, de l'infrastructure, ou du réseau.

Un DBA efficace travaille en collaboration, pas en silo.

7 Humilité technique

Même avec de l'expérience, on peut se tromper.

Un DBA senior doit savoir remettre en question son diagnostic et accepter d'escalader ou de demander un second avis si nécessaire.

8 Anticipation

Le rôle du DBA n'est pas seulement de réparer, mais d'éviter les incidents.

Surveiller les tendances, prévoir la croissance des données, détecter les signaux faibles fait partie des qualités clés.

9 Gestion du stress et des astreintes

Les environnements critiques impliquent parfois des interventions en dehors des heures normales.

Il faut être capable de gérer ces situations avec calme et professionnalisme.

9 Gestion du stress et des astreintes

Les environnements critiques impliquent parfois des interventions en dehors des heures normales.

Il faut être capable de gérer ces situations avec calme et professionnalisme.

10 Culture de l'amélioration continue

Après un incident, un DBA senior ne se contente pas de résoudre : il analyse la cause racine et met en place des actions pour que cela ne se reproduise pas.

SIMULATION D'ENTRETIEN DBA ORACLE

 **Recruteur : Bonjour, pouvez-vous vous présenter rapidement ?**

 **Toi :**

Je suis DBA Oracle avec une forte expérience en environnements critiques. J'ai l'habitude de gérer des bases fortement sollicitées, avec des exigences élevées en disponibilité, performance et fiabilité. J'interviens à la fois sur le RUN — supervision, incidents, optimisation — et sur l'amélioration continue : tuning SQL, gestion de la volumétrie, sécurisation des accès et industrialisation des procédures.

 **Recruteur : Qu'est-ce qui vous attire chez Fives Syleps ?**

Ce qui m'intéresse particulièrement, c'est que vos bases soutiennent des systèmes logistiques et industriels en production. Ici, une base lente ou indisponible ne gêne pas seulement un logiciel, elle peut bloquer des flux physiques réels. C'est exactement le type d'environnement critique dans lequel je souhaite intervenir, où le rôle du DBA a un impact direct sur l'activité opérationnelle.


 **Recruteur : Comment garantissez-vous la disponibilité d'une base 24/7 ?**

Je travaille sur trois axes : prévention, résilience et réactivité.

Prévention via la supervision proactive des ressources et de la volumétrie.

Résilience avec des sauvegardes fiables et testées, et si nécessaire des solutions de réplication type Data Guard.

Réactivité avec des procédures d'incident claires pour réduire au maximum les interruptions.

 **Recruteur : Une base devient lente mais le CPU est normal. Que faites-vous ?**

Je regarde immédiatement les attentes Oracle. Si le CPU est bas, le problème est souvent I/O ou verrouillage. J'analyse les wait events, les latences disque, puis les sessions actives et les verrous bloquants. L'objectif est d'identifier si la base attend le disque, un verrou ou une requête mal optimisée.

Recruteur : Un batch bloque la production le matin, comment réagissez-vous ?

Je vérifie si le batch est encore actif et s'il génère des verrous ou une forte activité I/O. Ensuite j'analyse son SQL et son plan d'exécution. À moyen terme, je propose de le replanifier hors pics, de l'optimiser ou de le découper en traitements plus petits pour éviter l'impact sur le temps réel.

Recruteur : Comment gérez-vous les verrous bloquants ?

J'identifie la session bloquante et la session bloquée, je regarde le type de verrou et le SQL exécuté. Si c'est une transaction anormalement longue, je coordonne avec l'équipe applicative avant toute action. Le kill session reste une solution de dernier recours, encadrée.

Recruteur : Que surveillez-vous en priorité sur une base critique ?

L'espace disque, la latence I/O, les sessions actives, les verrous, le top SQL, la génération de redo et les jobs en échec. L'idée est d'anticiper avant qu'un incident n'impacte la production.

Recruteur : Comment abordez-vous la performance SQL ?

J'utilise AWR et ASH pour identifier les requêtes dominantes, j'analyse les plans d'exécution, je vérifie les index et les statistiques. J'essaie toujours d'améliorer la performance sans perturber la stabilité globale.

Recruteur : Quelle est votre approche de la sécurité des bases ?

Principe du moindre privilège, gestion rigoureuse des rôles, audit des accès sensibles et traçabilité des opérations. Dans un contexte industriel, la protection des données de traçabilité est essentielle.

Recruteur : Comment travaillez-vous avec les développeurs ?

En collaboration. J'aide à analyser les requêtes lentes, je conseille sur la modélisation et j'anticipe les impacts volumétriques. L'objectif est d'éviter les problèmes de performance dès la conception.


 **Recruteur : Quelles sont selon vous les erreurs qu'un DBA ne doit jamais faire ?**

Intervenir en production sans analyse, négliger les sauvegardes, tuer une session sans comprendre, lancer des opérations lourdes en pleine charge ou ignorer les signaux faibles. En production critique, chaque action doit être mesurée et réversible.

 **Recruteur : Quelles qualités humaines sont importantes pour ce poste ?**

Le sang-froid, la rigueur, la capacité à prioriser et à communiquer clairement en situation d'incident. Un DBA doit rassurer autant qu'il résout.

Simulation — Recruteur technique très
pointu


 **Recruteur : Quelle est la différence entre un problème CPU et un problème I/O dans Oracle ?**

 **Toi :**

CPU → sessions actives en calcul, temps DB dominé par CPU

I/O → temps DB dominé par wait events liés au disque (sequential read, scattered read)

Je regarde toujours le ratio DB time CPU vs wait events dans AWR.

 **Recruteur : Pourquoi une base peut être lente alors que le serveur est peu chargé ?**

Parce qu'Oracle peut attendre : I/O, verrous, log file sync, ou contention mémoire.

Le système peut sembler "calme" mais la base est en attente.

 **Recruteur : Expliquez "log file sync"**

Attente liée à la validation des commits.

Si élevée : redo logs trop petits, disques lents, trop de commits fréquents.

Recruteur : Comment diagnostiquez-vous un problème de contention ?

Je regarde les waits : enq: TX, buffer busy waits, latch contention.
Puis j'identifie l'objet et le SQL en cause.

Recruteur : Qu'est-ce que le bind peeking peut provoquer ?

Mauvais plan d'exécution si la première valeur de bind n'est pas représentative.
Oracle peut réutiliser un plan inadapté.

Recruteur : Comment éviter ORA-01555 ?

UNDO suffisant, undo_retention adapté, éviter requêtes longues sur données volatiles.

Recruteur : Différence entre Data Guard et RAC ?

RAC = haute disponibilité locale + scalabilité
Data Guard = reprise après sinistre entre sites

Recruteur : Que faites-vous si les redo switch sont trop fréquents ?

J'augmente la taille des redo logs pour réduire la fréquence et éviter checkpoint pressure.

Recruteur : Comment analyser un SQL qui consomme 40% du DB time ?

AWR pour identifier
DBMS_XPLAN pour plan
Vérifier stats, index, cardinalités
Tester amélioration hors prod

Recruteur : Pourquoi un index peut ne pas être utilisé ?

Stats incorrectes, mauvaise sélectivité, fonction sur colonne, hint manquant, ou coût estimé trop élevé.

Recruteur : Que surveillez-vous pour anticiper une saturation disque ?

Croissance tablespaces, archive log rate, I/O latency, espace FRA.

Recruteur : Comment gérez-vous un deadlock ?

Oracle en tue une automatiquement.

Je récupère le trace, identifie SQL, corrige la logique applicative ou l'ordre d'accès aux tables.

Recruteur : Expliquez l'impact d'un full table scan en prod

Forte I/O, concurrence avec transactions, cache buffer saturé → ralentissement global.

Recruteur : Différence entre shared pool et buffer cache ?

Shared pool = SQL, plans, metadata

Buffer cache = blocs de données

Recruteur : Pourquoi un hard parse est coûteux ?

Compilation SQL + latch shared pool + CPU.

Beaucoup de hard parses = contention et perte performance.

 **Recruteur : Qu'est-ce qu'un AWR vous apporte qu'ASH seul ne donne pas ?**

AWR donne la vue historique agrégée.

ASH donne le détail échantillonné session par session.

 **Recruteur : Comment gérez-vous la croissance des tables logistiques ?**

Partitionnement, purge/archivage, compression si pertinent.


 **Recruteur : Que faites-vous avant de supprimer un index ?**

Vérifier son usage via AWR/ASH et plans SQL.

Jamais supprimer sans preuve.

 **Recruteur : Pourquoi un commit trop fréquent peut poser problème ?**

Génère beaucoup de redo et augmente log file sync.

 **Recruteur : Dernière question : c'est quoi pour vous un bon DBA en environnement industriel ?**

Un DBA qui anticipe les incidents, stabilise vite en cas de problème, et comprend que derrière la base il y a des opérations réelles à ne jamais bloquer.



LEXIQUE DES TERMES ORACLE

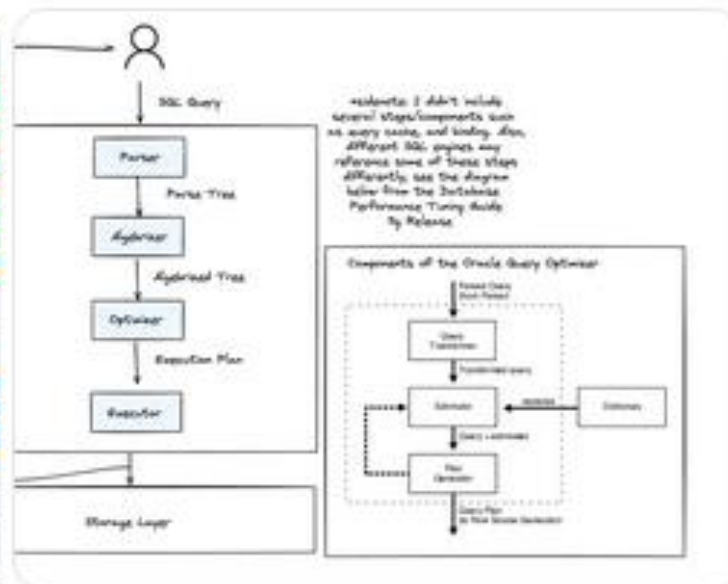


1. Concepts généraux Base de Données



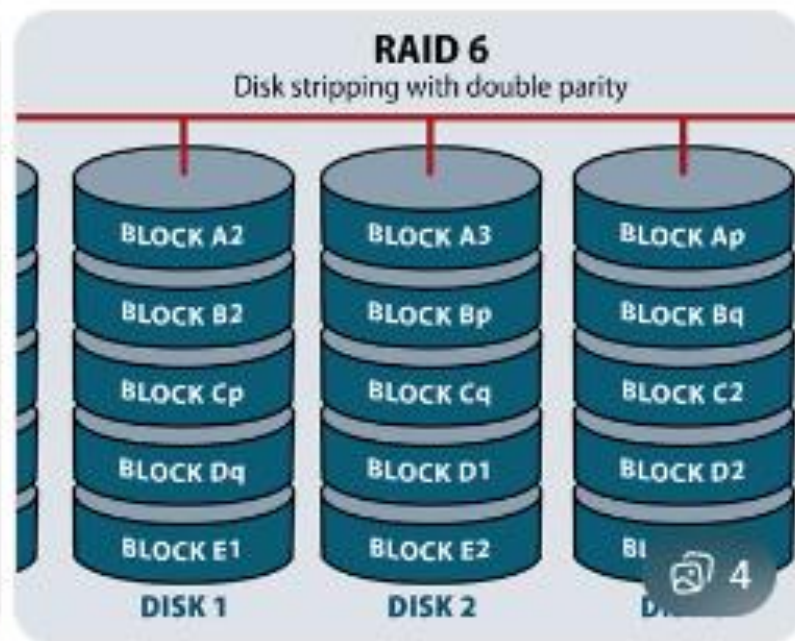
Terme	Signification	Explication simple
DBMS	Database Management System	Logiciel qui gère une base de données (Oracle, MySQL...)
SGBD	Système de Gestion de Base de Données	Version française de DBMS
Instance	Instance Oracle	Processus + mémoire qui font tourner la base
Database	Base de données	Fichiers physiques contenant les données
Schema	Schéma	Ensemble d'objets appartenant à un utilisateur
Transaction	Transaction	Ensemble d'opérations validées par un COMMIT
Commit	Validation	Rend les modifications permanentes
Rollback	Annulation	Annule les modifications non validées
Session	Connexion utilisateur	Processus actif connecté à la base

⚙️ 2. Performance & Diagnostic



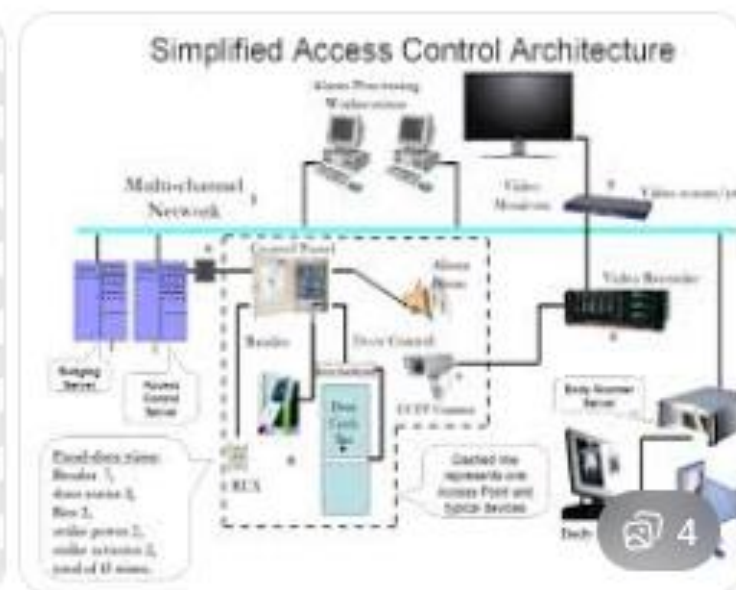
Terme	Signification	Rôle
AWR	Automatic Workload Repository	Historique de performance Oracle
ASH	Active Session History	Détail des sessions actives
ADDM	Automatic Database Diagnostic Monitor	Analyse automatique des problèmes
Wait Event	Événement d'attente	Indique pourquoi Oracle attend
DB Time	Temps base	Temps total passé par la base
Execution Plan	Plan d'exécution	Stratégie choisie par Oracle pour une requête
Full Table Scan	Lecture complète table	Oracle lit toute la table
Index Scan	Lecture via index	Accès rapide par index
Bind Peeking	Analyse première valeur	Peut influencer le plan SQL
Hard Parse	Compilation SQL	Création d'un nouveau plan d'exécution
Soft Parse	Réutilisation SQL	Utilise un plan déjà existant

3. Stockage & I/O



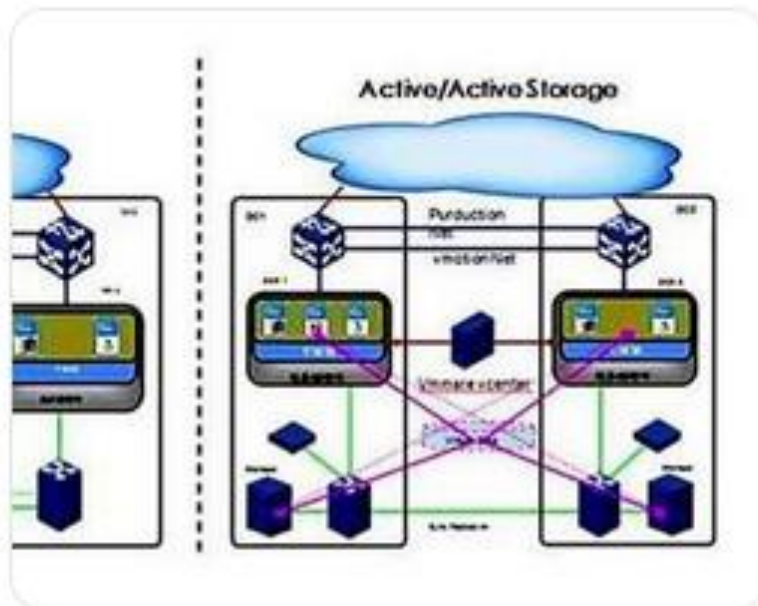
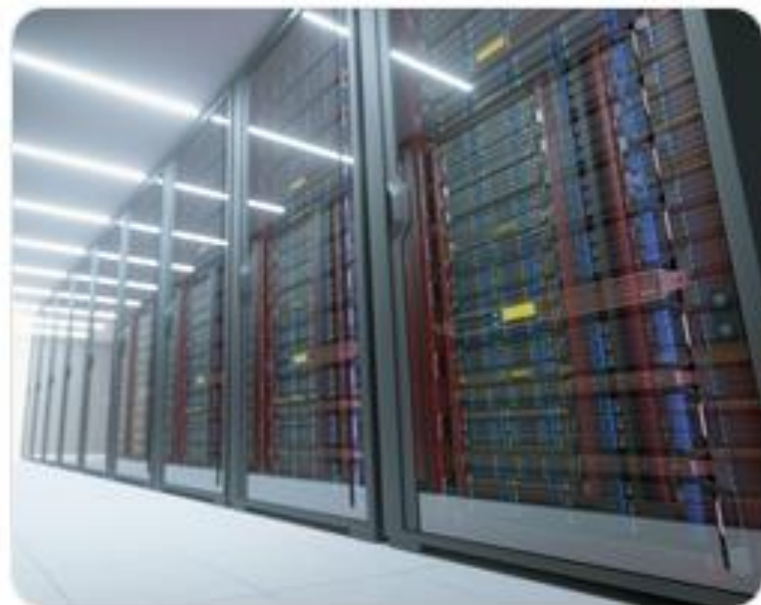
Terme	Signification	Rôle
Tablespace	Espace logique	Conteneur de stockage Oracle
Datafile	Fichier de données	Fichier physique sur disque
Redo Log	Journal redo	Trace toutes les modifications
Redo Switch	Changement de redo	Passage au journal suivant
Archive Log	Redo archivé	Copie des redo pour recovery
UNDO	Données d'annulation	Permet rollback et cohérence lecture
TEMP	Tablespace temporaire	Tri, jointures, opérations lourdes
I/O Latency	Latence disque	Temps d'accès au stockage

4. Sécurité & Accès



Terme	Signification	Rôle
User	Utilisateur	Compte Oracle
Role	Rôle	Groupe de privilèges
Privilege	Privilège	Droit d'accès ou action
Audit	Audit	Traçabilité des actions
Encryption	Chiffrement	Protection des données
TDE	Transparent Data Encryption	Chiffrement des fichiers Oracle

5. Haute Disponibilité & Sauvegarde



Terme	Signification	Rôle
RMAN	Recovery Manager	Outil sauvegarde Oracle
Backup	Sauvegarde	Copie des données
Restore	Restauration	Remise en état après perte
Recovery	Récupération	Rejoue les redo
PRA	Plan de Reprise d'Activité	Redémarrage après sinistre
Data Guard	Réplication Oracle	Copie base vers site secondaire
Standby DB	Base secondaire	Copie prête à prendre le relais
RAC	Real Application Clusters	Plusieurs serveurs pour une même base
Failover	Bascule automatique	Passage vers système secondaire

6. Mémoire Oracle

Terme	Signification	Rôle
SGA	System Global Area	Mémoire partagée Oracle
PGA	Program Global Area	Mémoire privée des sessions
Buffer Cache	Cache données	Blocs lus depuis disque
Shared Pool	Cache SQL	Plans SQL et métadonnées
Library Cache	Cache requêtes	Stocke SQL compilé

7. Objets & Optimisation SQL

Terme	Signification	Rôle
Index	Structure accès rapide	Accélère les lectures
Histogram	Statistique avancée	Aide l'optimiseur
Partitioning	Partitionnement	Découpe une table volumineuse
Cardinality	Cardinalité	Nombre estimé de lignes
CBO	Cost Based Optimizer	Optimiseur basé sur les coûts

8. Infrastructure & Cloud

Terme	Signification	Rôle
AWS	Amazon Web Services	Cloud Amazon
OCI	Oracle Cloud Infrastructure	Cloud Oracle
IaaS	Infrastructure as a Service	Serveurs virtuels
SAN	Storage Area Network	Réseau de stockage
VM	Machine virtuelle	Serveur virtualisé

TOP 50 des termes Oracle/DBA

⚡ Performance & Diagnostic (priorité absolue)

Terme	À retenir absolument
AWR	Historique performance base
ASH	Sessions actives en détail
ADDM	Diagnostic automatique Oracle
DB Time	Temps total passé en base
Wait Event	Pourquoi Oracle attend
Execution Plan	Stratégie d'exécution SQL
Full Table Scan	Lecture complète table
Index Scan	Accès via index
Hard Parse	Compilation SQL coûteuse
Soft Parse	Réutilisation plan SQL
Bind Peeking	Plan basé sur 1ère valeur bind
Cardinality	Nombre de lignes estimé
Histogram	Stats avancées pour skew
CBO	Optimiseur basé sur le coût

Stockage & I/O

Terme	À retenir absolument
Tablespace	Conteneur logique de données
Datafile	Fichier physique de données
Redo Log	Journal des modifications
Redo Switch	Changement de fichier redo
Archive Log	Redo sauvegardé pour recovery
UNDO	Annulation & cohérence lecture
TEMP	Espace temporaire SQL lourd
I/O Latency	Temps réponse disque
FRA	Zone fichiers recovery

Haute disponibilité & Sauvegarde

Terme	À retenir absolument
RMAN	Outil sauvegarde Oracle
Backup	Copie de sécurité
Restore	Restauration fichiers
Recovery	Rejoue les redo
PRA	Reprise après sinistre
Data Guard	Réplication base distante
Standby DB	Base de secours
RAC	Cluster Oracle multi-nœuds
Failover	Bascule vers secours
Checkpoint	Synchronisation mémoire/disque

Sécurité & Accès

Terme

À retenir absolument

User

Compte base

Role

Groupe de privilèges

Privilege

Droit d'accès

Audit

Traçabilité actions

TDE

Chiffrement données Oracle

Mémoire Oracle

Terme

À retenir absolument

SGA

Mémoire partagée Oracle

PGA

Mémoire privée session

Buffer Cache

Cache blocs données

Shared Pool

Cache SQL & metadata

Library Cache

SQL compilé stocké

Exploitation & Incidents

Terme

À retenir absolument

Session

Connexion active

Lock

Verrou Oracle

Deadlock

Blocage mutuel sessions

Latch

Verrou mémoire interne

Log File Sync

Attente validation commit

Snapshot Too Old

ORA-01555 UNDO insuffisant

Infra & Architecture

Terme

À retenir absolument

Instance

Processus Oracle + mémoire

Database

Fichiers physiques

Schema

Objets d'un user

Partitioning


Découpage grande table

Les termes à connaître par coeur dans
l'écosystème Oracle

PRIORITÉ 1 — DISPONIBILITÉ

Objectif : zéro arrêt des flux

- RMAN = sauvegarde
- Restore/Recovery = restauration + redo
- Redo Logs = journal des transactions
- Archive Logs = redo pour recovery
- Data Guard = réplication PRA
- RAC = haute dispo locale
- Failover = bascule secours
- Checkpoint = synchro mémoire → disque

 *Phrase clé :*

“La base ne doit jamais être un point de blocage opérationnel.”

⚡ PRIORITÉ 2 — PERFORMANCE

Objectif : temps réel fluide

- AWR = historique perf
- ASH = sessions actives
- DB Time = temps total base
- Wait Events = pourquoi Oracle attend
- Execution Plan = stratégie SQL
- Full Table Scan = lecture complète
- Index Scan = lecture ciblée
- Hard Parse = compilation coûteuse
- Bind Peeking = plan biaisé possible
- Cardinality = estimation nb lignes

💬 *Phrase clé :*

“Je cherche toujours si Oracle travaille... ou s’il attend.”

PRIORITÉ 3 — STOCKAGE / I-O

- Tablespace = conteneur logique
- Datafile = fichier physique
- UNDO = rollback & cohérence lecture
- TEMP = tris et requêtes lourdes
- Redo Switch = changement journal
- I/O Latency = lenteur disque = base lente
- FRA = zone fichiers recovery

 *Phrase clé :*

“CPU bas + lenteur = souvent un problème disque.”



PRIORITÉ 4 — SÉCURITÉ

- User = compte
- Role = groupe droits
- Privilege = autorisation
- Audit = traçabilité
- TDE = chiffrement fichiers



Phrase clé :

“Principe du moindre privilège et traçabilité des accès.”

PRIORITÉ 5 — MÉMOIRE

- SGA = mémoire partagée
- PGA = mémoire session
- Buffer Cache = blocs données
- Shared Pool = SQL + plans
- Library Cache = SQL compilé

 *Phrase clé :*

“Hard parses excessifs = contention Shared Pool.”



PRIORITÉ 6 — INCIDENTS CLASSIQUES

Problème	Cause probable
Base lente CPU bas	I/O ou verrous
ORA-01555	UNDO insuffisant
Log file sync élevé	Redo logs petits/disque lent
Sessions bloquées	Transaction longue
Batch qui bloque	Verrous + I/O
Requête devenue lente	Plan changé

