

## Solution Brief

# Solutions for Proprietary and Oracle Legacy applications

Performance improvement, QoS stability and Data Leakage Prevention

## Challenges

- Improve performance in Proprietary and Oracle Legacy applications where source code optimization is not possible → with **ActiveBase Performance**
- Prevent Personal Information data leakage in Proprietary and Oracle Legacy applications that has not been developed with the new security compliance requirements → with **ActiveBase Security**

## Customer Experience

A large Telecom company uses an old Siebel application on an Oracle8.0 database version. Oracle experts have been assigned to reduce transactions response times from 1.5 hours to less than 20 minutes. The result of their work was a recommendation to rewrite several SQL requests.

As the customer could not change the source code, recommendations were not applied UNTIL **ActiveBase Performance** was installed. Within a day, the application was configured to use ActiveBase server and a few rules were defined that dynamically substituted the inefficient SQL requests with the experts' recommendations, without touching the source code.

The result: **Performance was improved and response time accelerated by over 200%!**

## ActiveBase Performance Solution

Challenge:

Proprietary and Oracle Legacy applications do not allow changing the source code, whilst the most effective performance improvement to long running requests, batches and cycles is achieved by rewriting the SQL source code.

As performance degrades, the only option left for performance improvement is buying more hardware and software licenses...

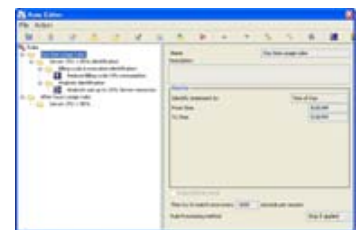
The Solution:

**ActiveBase Performance** user-defined rules, built on unique SQL\*Net Proxy, enables to transparently intercept long running SQL requests and apply various SQL optimization techniques, including rewrites or adding Oracle Hints, with no changes to application source code or databases!!



**Secure application's forms and canned reports access to sensitive and personal information without touching source code!**

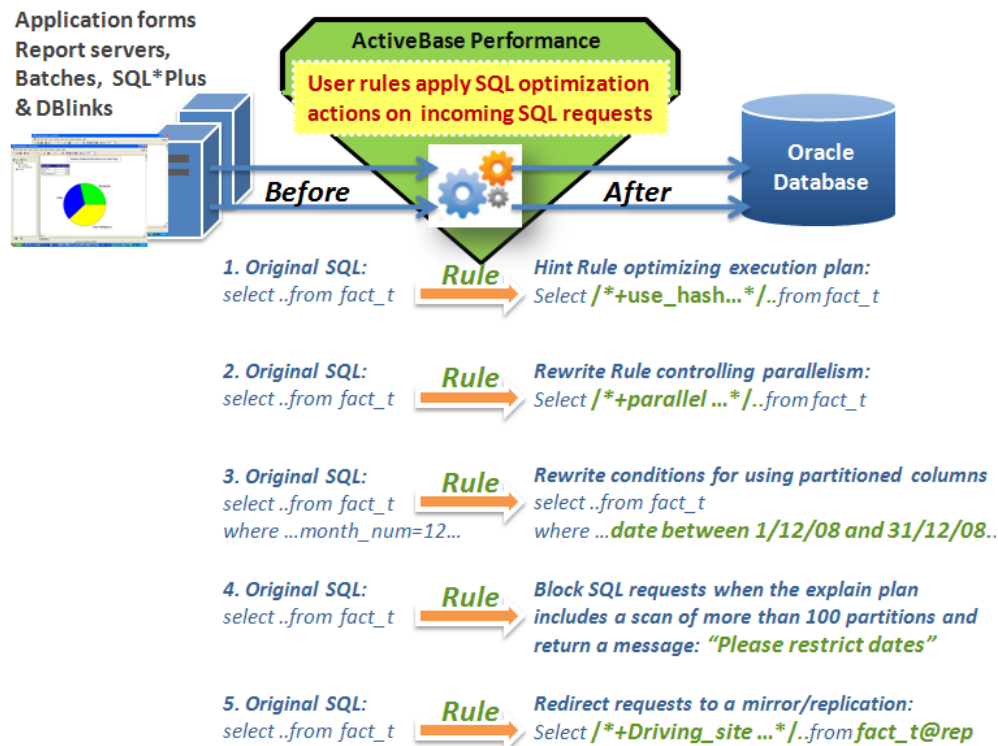
**Improve performance by tuning long-running SQL requests transparently**



Rule Management



**ActiveBase Performance** SQL optimization rules result in up to x100 faster response time.



**ActiveBase Performance** is transparently installed without changing applications or touching databases

## ActiveBase Security Solution

Challenge:

Protecting sensitive and personal information on all production environments has become a corporate imperative for all organizations due to the slew of global regulations governing data privacy.

1. Proprietary and Oracle Legacy applications often do not manage appropriately access to personal information and therefore do not comply with existing regulations. Expanding existing application security is not possible, since source code changes are not possible!
2. Restricting access to personal information from privileged users (as supplied for example by Database Vault or VPD), is not supported in older Oracle versions.

The Solution:

**ActiveBase Security** offers a new approach. Built on a unique SQL\*Net Proxy, supporting Oracle8.0 - 11g, it transparently intercepts SQL requests coming from application screens, interfaces or packaged reports and applies Security Rules for:

1. Blocking unauthorized access to production databases while returning a customized message back to the user
2. Auditing database usage and access to sensitive objects
3. Masking sensitive information on-the-fly, preventing access by unauthorized personnel and IT developers in production, testing and development environments

## Benefits

- ✓ Save on expensive application upgrades, hardware purchases and database software updates.
- ✓ Improve response time from hours to minutes using ActiveBase SQL optimization rules.
- ✓ Implement your DBAs and performance experts' recommendations, where otherwise not possible!
- ✓ Prolong the lifespan of your existing applications.



4. Limiting access to information by automatically adding 'where' clause and column access restrictions.
5. Transparent encryption/decryption of sensitive and personal data.

## Benefits

- ✓ Prevent leakage of Sensitive and Personal Information from reporting or DBA Tools accessing the production environment.
- ✓ Allow selective access to personal information based on time-of-day, specific screens, users, modules or specific Security Officer grants
- ✓ Enforce application security policies across applications and tools accessing production database.
- ✓ Apply database security, without changing database configurations or data!
- ✓ Secure SQL access to databases even when code source is not available!
- ✓ Traceability: ActiveBase audits, controls and alerts on access to SPI or changing user grants and account information.
- ✓ Audit, control or block all users NOT even passing through AB\*Security

All security rules can be applied selectively, based on condition values including access from specific applications, modules, screens, requests and users.

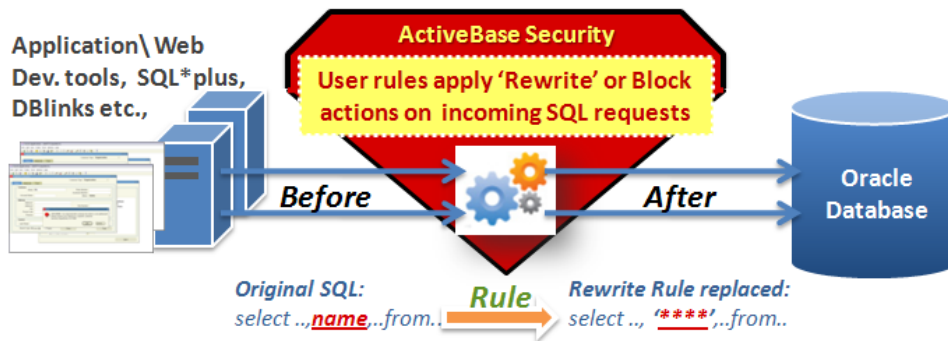
Different security rules can be applied to internal or external application users, DBAs and developers.

*For example, users can define in minutes a rule in ActiveBase Security rule management GUI that masks all SSN (Social Security Number) information retrieval requests executed from specific production application screens and from all canned reports. Rule is applied without touching application code or database configurations*

Rules are powerful and can have a broad scope, enabling a single rule to mask or block all SQL requests that retrieve personal information.

Adding or changing security rules are applied immediately, enabling users to work through the application screens while ensuring compliance and preventing leakage of sensitive and personal information.

Rules apply across databases and applications, with NO CHANGES TO THE APPLICATIONS OR THE DATABASE!



Masking Rules:	Scrambling Rules:	Hiding Rules:																														
Original SQL: <code>Select name,..from..</code>	Original SQL: <code>select name,..from..</code>	Original SQL: <code>select ..,name,..from..</code>																														
After Rule: <code>Select substr(name,1,2)  '***'</code>	After Rule: <code>Select scrmbl(name)</code>	After Rule: <code>select ..,..from..</code>																														
<table border="1"> <tr><th>Name</th><th>Name</th></tr> <tr><td>Tiger</td><td>Ti***</td></tr> <tr><td>Nelson</td><td>Ne***</td></tr> <tr><td>Rogers</td><td>Ro***</td></tr> <tr><td>Rosen</td><td>Ro***</td></tr> </table>	Name	Name	Tiger	Ti***	Nelson	Ne***	Rogers	Ro***	Rosen	Ro***	<table border="1"> <tr><th>Name</th><th>Name</th></tr> <tr><td>Tiger</td><td>Bell</td></tr> <tr><td>Nelson</td><td>Cave</td></tr> <tr><td>Rogers</td><td>Lennon</td></tr> <tr><td>Rosen</td><td>Lenin</td></tr> </table>	Name	Name	Tiger	Bell	Nelson	Cave	Rogers	Lennon	Rosen	Lenin	<table border="1"> <tr><th>Name</th><th>Name</th></tr> <tr><td>Tiger</td><td></td></tr> <tr><td>Nelson</td><td></td></tr> <tr><td>Rogers</td><td></td></tr> <tr><td>Rosen</td><td></td></tr> </table>	Name	Name	Tiger		Nelson		Rogers		Rosen	
Name	Name																															
Tiger	Ti***																															
Nelson	Ne***																															
Rogers	Ro***																															
Rosen	Ro***																															
Name	Name																															
Tiger	Bell																															
Nelson	Cave																															
Rogers	Lennon																															
Rosen	Lenin																															
Name	Name																															
Tiger																																
Nelson																																
Rogers																																
Rosen																																

*ActiveBase Security is transparently installed without changing applications or touching databases*

[www.active-base.com](http://www.active-base.com)

400-00101-005 | 02/09 | © 2009 ActiveBase, Ltd. All rights reserved. All other third-party trademarks are the property of their respective owners.

Page 3

