

*rego*University 2017

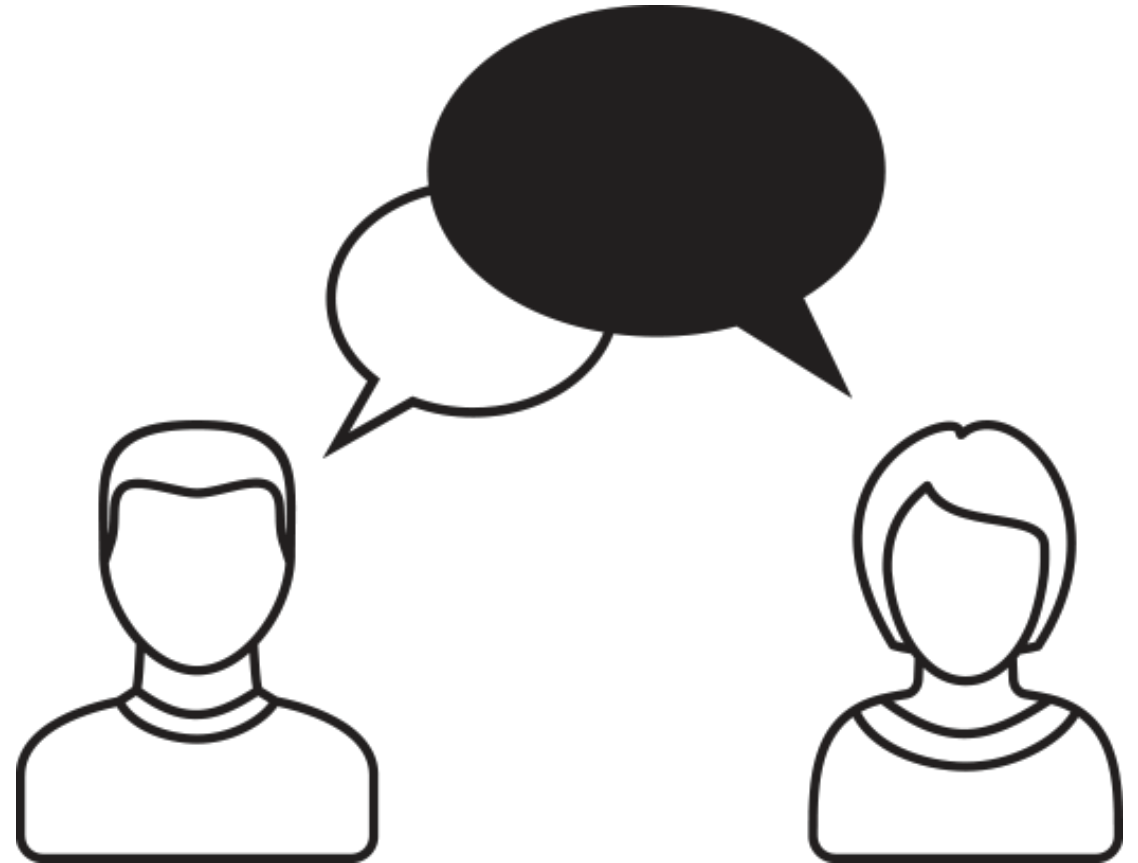
System Performance | Tips and Tricks

Your Guide: Josh Leone, Vipin Chouhan



Introductions

- Take 5 Minutes
- Turn to a Person Near You
- Introduce Yourself



Agenda

- About the Performance Team
- Performance Elements
- Performance Process
- Case Studies
- Case Study Solutions
- Quick Monitoring Technique

Rego Performance Team

- **Kelly Limberg**
 - Former CA SWAT 18 year CA PPM Experience
- **Steve Seaney**
 - Managed multiple 40K user instances of CA PPM
- **Josh Leone**
 - Former CA SWAT 18 year CA PPM Experience
- **David Matzdorf**
 - Rego Chief Architect / SQL Tuning Expert
- **Karunakaran Nadarajah (Karuna)**
 - Developer / Database and SQL Tuning Expert
- **Chris Schaffer**
 - Developer / Database and SQL Tuning Expert
- **Vipin Chouhan**
 - Developer / Offshore global coverage

Rego is the largest specialized CA PPM consulting firm in the world – we know CA PPM.

Rego has staff that have "lived" management, support and development roles for large instances vs. just consulted to large organizations.

Rego has a robust performance tuning practice that takes a waved approach to performance analysis and tuning - starting with application settings down to tuning specific SQL execution plans within the CA PPM database.

Performance Data Exists In Many Data Sources

- Holistic performance approach – Focused on end-user experience
- There are a variety of sources providing key information on performance.
 - Many are only available for On Premise

Log & Config Files

- App Logs
- BG Logs
- App Access Logs
- Process engine

Infrastructure

- Diagrams
- Confirm Cores
- Confirm Memory
- Confirm IO
- Confirm Network
- GC Performance

Database Reports

- Oracle advantages
- Deadlocks
- Waits
- Physical IO
- Cache Memory
- Long sessions
- Top SQL by CPU
- Top SQL by IO
- Table fragmentation
- Table configuration

Key Levers To Improve Performance

- Items driving performance are typically resolved by adjustments to the base infrastructure, application deployment, and the application configuration.

Base Infrastructure

- VM Architecture
- Database Cores
- Database Memory
- Shared usage

Application Deployment

- JVM Architecture
- JVM Memory
- App Server Maintenance
- Database Settings

Application Configuration

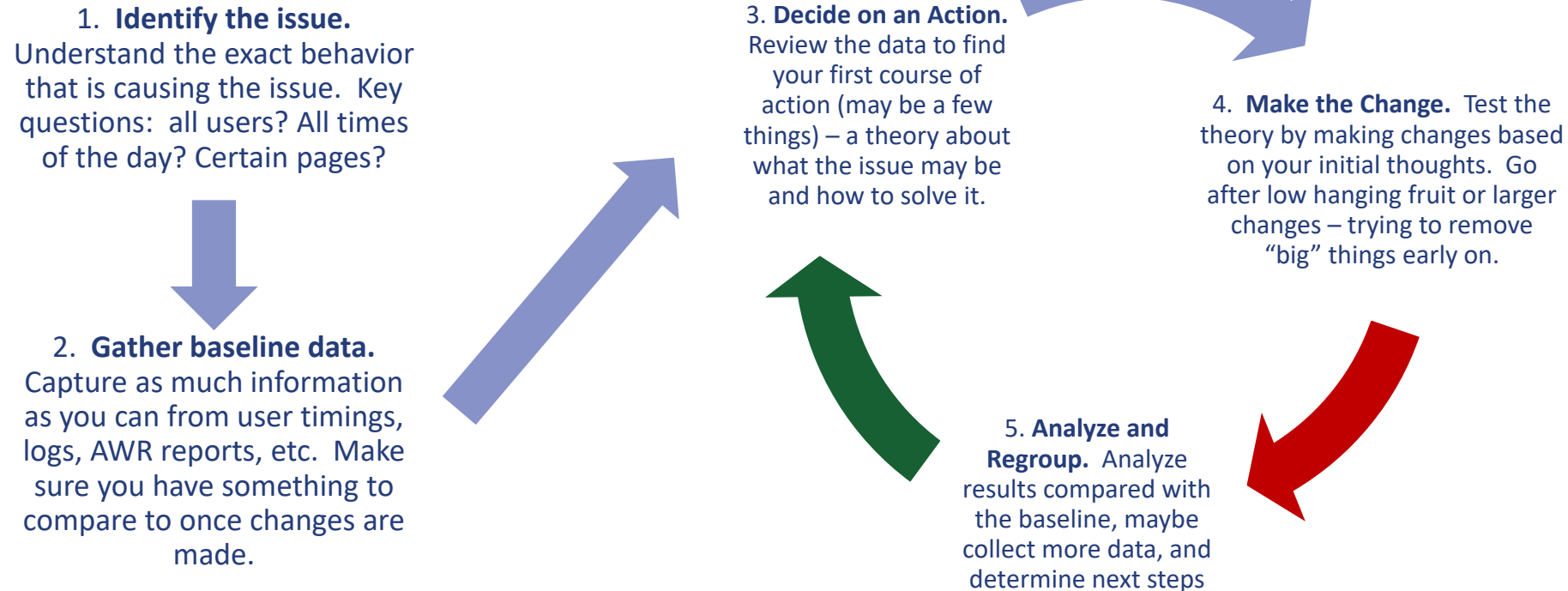
- Jobs
- Pages
- Security
- Settings
- Code Optimization
- Data Limits

On Premise Only

Performance Improvement Process

Always start performance work with the facts – without understanding exactly what you are trying to fix – you will never know when to declare success.

After you have the facts, addressing performance issues is an iterative process - you start with the low hanging fruit and move through additional layers over time.



Case Study: Hidden Long Actions

Industry	Financial Services Organization Asset Management, Business Processing
Facts	140 GB Oracle database 15,000 Active Users 20,000 Active Projects
Overview	End users ran resource planning portlets without training. Out of curiosity, they launched and examined data and hit run-buttons repeatedly.
Details	Several BOE reports ran for over two hours without filters. One user ran a portlet fifteen times in ten minutes, strained application servers doing data crunching and cascaded slowdowns throughout the environment.

Diagnosis And Results

Despite having a well-sized system, long-running reports and portlets—as well as frequent and unknown runs—caused server slow downs.

The Customer was surprised to find out users ran reports without filters, and it was typical to generate project status reports for over 6,000 projects by mistake.

Rego Performance Technician came in with recommendations for healthy report deployment, required parameters to shore up the system, and training solutions to solve the problem.

Case Study: Hidden Long Actions

- Review Report history

- Using BusinessObjects instance manager, look at historical report instances for any taking over a few minutes
- Educate users to schedule reports off-hours
- Report history in the CA PPM application access logs for immediate run reports (Rego can help parse logs)

- Lockdown Reports and portlets

- So they cannot be run wide open
 - Set required parameters on both reports and portlets
 - Required filter parameter will remove the Show All button
 - Make sure the initial or no filter response is adequate
 - Set Don't show results until I filter

Case Study: Object List Page/Portlets

- Attribute value protection
 - Set to second option "use secure subpages"
 - Don't forget Portlets based off of stock objects (non-NSQL)
- Aggregation rows
 - Largest performance impact on timescale views
 - Limit on portlets that can return many rows
- Use Lists for Navigation and Portlets for reporting
 - Locked down version of the project list page
 - Use for navigation only with the fastest setting.
 - Add to application menu as Project list Navigation Only

Case Study: Page And Timeslice Configuration

Industry	Manufacturing Organization Chemicals, Fibers, and Plastics
Facts	90 GB MSSQL Database 500 Active Users 14,000 Active Projects
Overview	Rather than deleting unneeded sections of OOTB project properties pages or using the risk attribute, the customer unknowingly added seconds to the project list page displays via inappropriate field removal.
Details	Technicians uncovered a significantly undersized system, as well as a background processes generating 10 million+ rows redundant /orphan access rights. We advised customer to contract for appropriate system capacity and clean up redundant /orphan rights.

Diagnosis And Results

Examining average click and display times, Rego discovered a 4-second display rate on the Project List page which signaled inappropriate conditions and had a cascading effect on list page performance. By removing display configurations and optimizing the page, Rego improved Project List responsiveness to 2.5 seconds.

Rego also discovered three-years-worth of unnecessarily maintained, redundant allocation data. By altering time splice tables and background jobs, they eliminated unneeded processing and improved system performance.

Case Study: Access Rights

- Are you bulk loading data via XOG?
 - Whenever items are bulk loaded an instance right is created
 - Ensure the XOG user has Global Rights
 - The instance rights can be safely removed (with Rego assistance for automation)
- Bloated security table and effect performance
 - Security checks occur all over, making them as efficient will help with general performance
 - Query the main CA PPM permissions table to look for candidates for purging instance rights rows
 - Typically if there is 15 to 20% of total rows we recommend clean up
- Clean up Methods
 - One time clean up via front end Via user's instance right page
 - If there are multiple millions of rows this may hang the front end and not complete
 - Rego can construct a delete statement which will remove the redundant instance rights in batches
 - The delete can be scripted as a CA PPM job as well

Case Study: Access Rights

Diagnostic Candidate Count Query

- Returns total number of access rights grouping by object code and permission code
- The object code can include custom objects as well as stock objects such as project and resource
- Results are limited to those having a count greater than 50,000

```
SELECT COUNT(*),
       U.USER_NAME,
       O.OBJECT_CODE,
       PR.PERMISSION_CODE
FROM CMN_SEC_ASSGND_OBJ_PERM P,
     CMN_SEC_USERS U,
     CMN_SEC_OBJECTS O,
     CMN_SEC_PERMISSIONS PR
WHERE P.PRINCIPAL_ID=U.ID
      AND P.OBJECT_ID = O.ID
      AND P.PERMISSION_ID = PR.ID
GROUP BY U.USER_NAME,
         O.OBJECT_CODE,
         PR.PERMISSION_CODE
HAVING COUNT(*) > 50000
ORDER BY 1 DESC
```

Case Study: Timeslice Configuration

- Count the rows to determine what has largest impact
 - Sample query ordered by count and grouped by request name
- Reduce ranges where possible
 - Only slice what is actually needed
- Review Timeslice recommendations from Rego U 2017 timeslice presentation

```
SELECT COUNT(*),  
       A.SLICE_REQUEST_ID,  
       (SELECT REQUEST_NAME  
        FROM PRJ_BLB_SLICEREQUESTS  
        WHERE ID = A.SLICE_REQUEST_ID)  
       REQUEST_NAME  
FROM PRJ_BLB_SLICES A  
GROUP BY SLICE_REQUEST_ID  
ORDER BY 1 DESC
```

Case Study: Portlets and Configuration

Industry	North American Auto Manufacturer Cars, Trucks and SUVs
Facts	150 GB Oracle 10,000 Users 20,000 Active Project
Overview	Database sessions lasted several minutes for select users, and several portlet queries, Webi reports, and pages required long wait-times due to poor SQL. Technicians uncovered queries causing major issues and advised the customer to contract for query tuning.
Details	Identifying poor performing queries is in scope of the subscription service however rewriting the queries is outside of the service but Rego can help

Diagnosis And Results

Particular users were frustrated with long wait times on select portlets and pages. Due to periodic system slow downs and oddly high database CPU, Rego diagnosed portlet and configuration tuning needs.

Technicians changed page configurations to only render with filters on and set default filters for the customer's portlets.

After eliminating long-running sessions that led to the system issues, and improving portlet and report speed, the customer's performance stabilized.

Case Study: Portlets Query Tuning

- Ensure defaults and required parameters are used
- Query design
 - Look to use precompiled sources where possible
 - Utilize new DWH with external data source
 - Avoid using a “NOT IN” in a query where clause, Use “NOT EXIST” instead
 - Consider using a stored procedure to populate a reporting table that is properly indexed (on premise only)
 - Use when there is just too much data for the query to be run efficiently
 - Write your portlet query against that table
 - Design your portlets/queries to enforce date range restrictions to prevent users from turning CA PPM into a personal Export-to-Excel data warehouse app
- Limit results when possible
 - For dynamic queries that can potentially return a large result set, be sure to limit the max number of rows returned if possible with:
 - Oracle: `ROWNUM < N`
 - SQL Server: `SELECT TOP N`

Case Study: Portlets Query tuning

- Avoid nested sub queries that return lots of data
 - If you must use them, be sure to parameterize them as well so as to minimize the number of rows needed
- Very Bad query example:
 - The below query could process millions of rows in the inner subquery
 - It would be more efficient to also include the where clause params in the subquery

```
SELECT MYSLICEDATA.PRJ_OBJECT_ID, SUM(MYSLICEDATA.SLICE)
FROM (SELECT * FROM PRJ_BLB_SLICES) MYSLICEDATA
WHERE MYSLICEDATA.SLICE_REQUEST_ID = 1
AND MYSLICEDATA.SLICE_DATE BETWEEN TRUNC(SYSDATE) -30 AND TRUNC(SYSDATE)
```

Case Study: Application Sizing

Industry	Insurance Property and Life insurance
Facts	150 GB Oracle 10,000 Users 20,000 Active Project
Overview	Despite a healthy database, during peak usage, users said the application was slow. “Out of Memory” Warnings popped up, and the application performed heap data dumps. JVMs spent significant time garbage collecting, with heaps as large as 1.5GB for the app and 1.0GB for the BG.
Details	Identifying the need for sizing is in scope of the subscription service however determining and implementing sizing is outside of the subscription service but Rego can help

Diagnosis And Results

Rego suspected sizing needs, due to CA PPM 13’s additional memory and core capacity requirements, and upon engagement found additional undersized symptoms.

Technicians increased JVM Application memory and BG memory. They enlarged database cores, and added JVM for XOG users, admins, and schedulers.

After recommending memory increases and watching GC performance when cores were limited, the combined approach stabilized the Dev environment, and was pushed production.

Case Study: Application Sizing

- Increase JVM memory
- RAM is cheap
 - Ensure the environment is adequately sized
 - 6 GB JVM for the APP service
 - 4 to 6 GB JVM for the BG Service.
- Scale beyond that by adding additional load balanced app services
- For smaller implementations you can scale without a load balancer by adding a dedicated XOG and or open workbench scheduler app instance

Case Study: Database Tuning

Industry	Manufacturing HVAC Equipment
Facts	100 GB Oracle 5,000 Users 10,000 Projects
Overview	The customer experienced frequent and persistent application slowness, with inconsistencies between the database CPU and application CPU.
Details	Technicians uncovered an undersized database cache and advised to contract for test phase coaching as they increased sizing (PGA and SGA).

Diagnosis And Results

Due to excessive database sessions lasting several minutes, session blocking, and high physical I/O, Rego diagnosed database tuning requirements.

Technicians moved stats Oracle jobs to night-time running, disabled the Oracle cron job, and rebuilt long-running SQL indexes.

After exposing the need for customary tuning and maintenance processes, as well as queries requiring contracted tuning rewrites, the customer enjoyed the return of a stable and consistent production environment.

Case Study: Database Tuning

- Increase database memory
- RAM is cheap
 - When additional memory is given to the database it will be used by the database buffer cache
 - For Oracle we like to see the buffer cache hit percentage over 99%
 - Any time it can fetch data from memory it is significantly faster than fetching data from disk
- The Oracle SGA advisor can only estimate so far
 - If you are grossly undersized it will not give an accurate estimation
 - For example if you have 6 GB allocated to the SGA and are in need of 24 GB the advisor will only give you advice up to 12 GB. When moving up the 12 it will advise once again to double 24.

Case Study: Database Tuning / Jobs

Scheduled job and database maintenance Window

- Ensure you know what your peak usage is for end-user traffic
 - This can be obtained by looking at the application access logs or last modify time on key tables such as prTimesheet
 - We typically see peak times of Friday afternoons and Monday late mornings, for North American customers
 - However if you user base is global the peak times will vary
- Determining the ideal time to run jobs is key for data availability maintenance and performance

Case Study: Database Maintenance

- Out-of-the-box database maintenance is insufficient
 - No out-of-the-box database maintenance for SQL Server
 - Oracle it is simply and analyze job
- We recommend..
 - collecting statistics and that tables and indexes are analyzed for fragmentation regularly
 - Then they can be re-indexed when there over a certain threshold i.e. 30% fragmented
 - Running a shrink operation on the tables will also aid in performance

Quick Monitoring

- Database Components
 - CPU
 - Memory
 - Sessions
 - Long Running SQLs
 - Invalid Database Objects
- Monitoring Logs
 - Issues / Errors
 - Usage monitoring using Access Logs

Questions?



*rego*University 2017

Let Rego be your guide.

Thank You For Attending regoUniversity

Instructions for PMI credits

- Access your account at pmi.org
- Click on **Certification**
- Click on **Maintain My Certification**
- Scroll down to **Report PDU's**
- Click on Course Training (or other appropriate category)
- Enter **Rego Consulting**
- Enter Activity- **Enter Name of Course**
- Enter **Description**
- Enter **Date Started**
- Enter **Date Completed**
- Provide Contact Person **Name of Person to Contact**
- Provide Contact E-Mail **E-Mail of Person to Contact**
- Enter Number of **PDU's Claimed** (1 PDU per course hour)
- Click on the **I agree this claim is accurate box**
- Click **Submit** button



Let us know how we can improve!
Don't forget to fill out the class survey.



Phone

888.813.0444



Email

info@regouniversity.com



Website

www.regouniversity.com