02: ♥♦ 8 real life examples of SAR technique for Java developers

Posted on September 7, 2014 by Arulkumaran Kumaraswamipillai — 1 Comment ↓

The open-ended questions give you a great opportunity to promote your strengths and accomplishments. SAR (**S**ituation-**A**ction-**R**esult) technique examples.

Example 1: Tuning performance

Situation: Performance problem where the application server had to be restarted every day.

9 tips to earn more | What can u do to go places? | **945+** members. LinkedIn Group. **Reviews**

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

open all | close all

- Ice Breaker Interview
  - 01: ♦ 15 Ice breake
  - 02: ♥♦ 8 real life ex
  - 03: ♦10+ Know you
  - 04: Can you think c
  - 05: ♥ What job inte
  - 06: ► Tell me abou
  - 07: ♥ 20+ Pre inter
- Core Java Interview C
  - Java Overview (4)
    - 01: ♦ ♥ 17 Java c

Action:

– Used JMeter to simulate the load conditions and reproduce the issue.
– Identified the cause of the problem to be leaking database connections due to not properly closing the connections under an exceptional scenario. Used the profiling tool "**VisualVM**". jvisualvm to detect memory leak
– Fixed the issue by closing the database connections in the finally block.
– Tuned the JVM settings and configured proper service timeouts.
– Load and endurance tested the fixed code with the load testing tool JMeter to confirm that the issue has been fixed.

Result: The application became a true mission critical 24×7 type with a much improved performance.

Example 2: Code quality

Situation: Java code that is hard to maintain and reuse. Changes to one module may break another module.

Action:

– Wrote unit tests with proper mock objects for the existing un-maintainable code.
– Introduced SonarQube to ascertain code coverage & get code quality metrics and fix the blocker, critical, and major severity items were fixed.
– Re-factored the code with OO concepts and design patterns in a test driven manner to improve maintainability.
– Large procedural style if/else statements were replaced with objects adhering to the Open-Closed design principle.
– Code duplication was eliminated with the help of SonarQube tool.
– Reran the unit tests to ensure that the functionality is not broken due to refactoring.

Result: The application became easier to maintain, extend, and reuse. The code coverage was increased from 27% to 76%.

Example 3: Quick wins

Situation: The financial service websites are ranked by an independent body, and this particular company's website that I was the technical lead for was ranked 23rd out of 31 possible companies that took part.

Action: I took the initiative with the collaboration of the business and technical leaders to launch a "**QuickWins**" project to improve the overall ranking of the website. An independent user experience consultant was hired to analyze and produce a report with 18 most important things that can potentially improve the user friendliness, look and feel, and ease of use of the overall website. Out of those 18 recommendations, 4 of them needed major design and development changes, and did not stack up well in the cost-benefit analysis. The remaining 14 recommendations were implemented within 3 months. The implementation was fast-tracked by adopting some of the agile development practices like iterative development, daily stand-up meetings, and regular catch-ups with the business.

Result: This initiative was a major success and the website ranking was improved from being 23rd to 12th. The management was very impressed, and the contributions were well noticed and rewarded. That was also one of my longest and rewarding contracts.

Example 4: Concurrency Management

Situation: The production ready application consumed very less CPU and response times were very poor due to heavy I/O operations like database read/write operations

Action:

– Monitored the CPU usage with Visual VM tool.
– Got a series of thread dumps, say 7 to 10 at a particular interval, say 5 to 8 seconds and analyzed those thread dumps by importing the thread dumps into “Samurai”.
– Paid attention to the blocked threads in red. Alternatively, VisualVM is handy for debugging deadlocks analyzing thread dumps.
– Fixed the concurrency issue by reducing the synchronization granularity in the code.
– The offending SQL statement was identified with a query planner and tuned.

Result: The response times were improved by 200% and CPU usage increased from 45% to 88%. The response times were halved.

Example 5: Design

Situation: The business wants to review the business rules and rapidly change them as the government compliance requirements like tax law changes.

Action:

- Used Drools (open-source rules engine) to externalize the rules to a rules database table.
- A Web based user interface was provided to the business to modify the rules in the database table.
- An approval process was designed to enable a separate user to authorize the change before it takes effect.
- Bulk changes to the rules can be carried out by the operations staff updating the rules table via SQL, but it needs to be tested and signed-off in a staging environment first.

Result: The business was empowered to react rapidly to the regulatory changes from the government. When the regulatory requirements change, the rules within the database tables can be modified in a controlled manner without requiring code changes in most cases.

Drools Tutorials – Non-trivial examples step by step

Example 6: Security

Situation: The SSL handshake issues whilst deploying the application to production environment.

Action:

- Use of keytool to verify certificates in the keystores and trusstores
- Exporting the missing certificates from the browser and importing it into the truststore.
- Enabling SSL debugging and using other handy tools like OpenSSL. The SSLPoke was a very handy Java program to debug SSL issues.

Result: SSL issues were resolved in a timely manner.

Example 7: Technical capabilities & experience

Situation: Develop a stand-alone TCP server that talks to 3000+ petrol pumps as part of the "pay at the pump" solution. The server also needs to integrate with 3+ back-end systems like payment gateway to settle the transactions, loyalty system to accrue & redeem rewards points, data warehouse system to produce 3 dimensional reports, and ESB(i.e. Enterprise Service Bus) to integrate with the ERP system.

Action:

- Developed the stand-alone server using a non blocking I/O framework known as the MINA.
- As a lead developer, had to liaise with internal staff & external vendors from the USA and South Africa. Attended many conferences and one on one sessions to gather functional & non-functional requirements.
- Reduced the complexity of the overall system by revising the architecture & reducing the number of

messages flowing among various systems from 89 to just 34.
- Built the system with low latency in mind. 13 Tips to write low latency applications in Java
- Fixed an **intermittent** load balancer issue.
- Managed a team of 7+ developers & liaised with 20+ other stake holders including external vendors, business analysts, systems team & the management.

Result: The high-visibility and low latency application was built on time and within budget with 75% unit test coverage.

Example 8: Facilitating process improvements

Situation: Tasked with transitioning the current project management process from waterfall to **agile**.

Action:

- Took on the role as a facilitator of the **Scrum** agile methodology, and empowered the teams to be more self-organizing
- Promoted Test Driven Development (TDD) and Behavior Driven Development (BDD) practices to increase the automated test coverage from 40% to 70%.
- Mentored the teams in shifting their mindsets from the big bang approach to iterative & incremental approach with the key focus of delivering a business value in each iteration.
- Trained staff on writing epics & stories, sprint planning, breaking down the stories into tasks, sizing the stories & tasks, conducting daily scrum & fort-nightly retrospective sessions, velocity points, progress tracking charts like burn-down & burn-up, etc

Result: A customized & practical agile process was embraced by the team that worked well for them without all

the rigid rules & rituals, but encompassing the core agile values and principles that added value to the business.

Key Take Away

Sell your strengths and well-rounded abilities to add value to the business. The key traits are: **1)** passion **2)** strong technical skills, **3)** necessary soft skills, and **4)** **right attitude/**.

So, don't get annoyed when you are technically grilled or challenged. If you don't know or in the wrong accept it. Show your enthusiasm for the job and the chosen career path.

Popular Posts

♦ 11 Spring boot interview questions & answers

852 views

♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

823 views

18 Java scenarios based interview Questions and Answers

447 views

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

400 views

♦ 7 Java debugging interview questions & answers

311 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

301 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

291 views

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

285 views

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

263 views

8 Git Source control system interview questions & answers

215 views

Bio | Latest Posts

Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.

About Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.

‹ 8 Java Annotations interview Questions and Answers

More Java annotations interview questions & answers ›

Posted in Ice Breaker Interview Q&A, Job Interview Ice Breaker Q&A Essentials

One comment on "02: ♥♦ 8 real life examples of SAR technique for Java developers"

Uma says:

September 27, 2016 at 1:25 am

This is an awesome article. Very informative and thought provoking. Thank you:)

Reply

Leave a Reply

Logged in as geethika. Log out?

Comment

Post Comment

- 08: Hibernate au
- 09: Hibernate en
- 10: Spring, Java
- 11: Hibernate de
- 12: Hibernate cu
- AngularJS (2)
  - ♥ 8 AngularJS in
  - More Angular JS
- Git & SVN (6)
  - ♥ Git & Maven fo
  - ♥ Merging Vs rel
  - ♥ Understanding
  - 6 more Git interv
  - 8 Git Source cor
  - Setting up Cygw
- JMeter (2)
  - ♥ JMeter for test
  - ♦ JMeter perform
- JSF (2)
  - JSF interview Q&
  - More JSF intervi
- Maven (3)
  - ♥ Git & Maven fo
  - 12 Maven intervi
  - 7 More Maven ir
- Testing & Profiling/Sa
  - Automation Testing
    - ♥ Selenium and
  - Code Coverage (2)
    - Jacoco for unit te
    - Maven and Cobe
  - Code Quality (2)
    - ♥ 30+ Java Code
    - ♦ Ensuring code
  - jvisualvm profiling (
    - 01: jvisualvm to
    - 02: jvisualvm to
    - 03: jvisualvm to
  - Performance Testir
    - ♥ JMeter for test
    - ♦ JMeter perform

As a Java Architect

Java architecture & design concepts interview Q&As with diagrams | What should be a typical Java EE architecture?

Senior Java developers must have a good handle on

80+ step by step Java Tutorials

Preparing for Java written & coding tests

open all | close all
- ♦ Complete the given
- Can you write code? (
- Converting from A to I
- Designing your classe
- Java Data Structures
- Passing the unit tests
- What is wrong with th
- Writing Code Home A
- Written Test Core Jav
- Written Test JEE (1)

How good are your…to go places?

open all | close all
- Career Making Know-
- Job Hunting & Resum

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ Java generics in no time ☀ Top 6 tips to transforming your thinking from OOP to FP ☀ How does a HashMap internally work? What is a hashing function?
☀ 10+ Java String class interview Q&As ☀ Java auto un/boxing benefits & caveats ☀ Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect

Non-Technical Know Hows

☀ 6 Aspects that can motivate you to fast-track your career & go places ☀ Are you reinventing yourself as a Java developer? ☀ 8 tips to safeguard your Java career against offshoring ☀ My top 5 career mistakes

Prepare to succeed

☀ Turn readers of your Java CV go from “Blah blah” to “Wow”? ☀ How to prepare for Java job interviews? ☀ 16 Technical Key Areas ☀ How to choose from multiple Java job offers?

Select Category

© Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.

↑