CISQ COMPLIANCE REPORT

Sample Application name – **Gridlock**

POWERED BY CAST SOFTWARE INTELLIGENCE

This is a sample report generated by CAST Software's Application Intelligence Platform ("AIP").

AIP has an outstanding prior performance within US Public Sector (DoD, Civilian, & State), the Global 2,000 largest corporations, Global & Federal System Integrators, Advisory Consultancies, and Independent Software Vendors ("ISV").

For more information about CAST Software solutions and to discuss how you may access a CISQ assessment for your own systems contact CAST at: publicsector@castsoftware.com.

CAST is a proud sponsor of CISQ's essential work since 2010.

Table of Contents

Introduction

This assessment is an effort to determine the overall quality of the Gridlock applications against CISQ rules and measure the overall health of the application. This assessment uses the CAST Application Intelligence Platform (AIP) (Version 8.3) to automatically scan the implementation of these applications to review the architecture, design, and code against current industry best practices and known design flaws that may impact performance.

CAST AIP applies over 1200 engineering checks based on standards and measurements developed by the Software Engineering Institute (SEI), International Standards Organization (ISO), Consortium for Information & Software Quality (CISQ), the Institute of Electrical and Electronics Engineers (IEEE) and the technology provider industry. The resulting analysis identifies specific flaws in the software and aggregates this information into metrics to objectively quantify the structural quality of the application.

CAST Appmarq is a benchmarking database of AIP analyses, and it compares application to peers in the same vertical and/or technology and ranks the application among its peers.

About CISQ – Consortium for Information & Software Quality

The Consortium for Information and Software Quality™ (CISQ™) is an IT industry leadership group comprised of IT executives from the Global 2,000, public sector, system integrators, outsourced service providers, and software technology vendors committed to introducing computable metrics standards for measuring software quality & size. CISQ is a neutral, open forum in which customers and suppliers of IT application software can develop an industry-wide agenda of actions for improving IT application quality to reduce cost and risk.

Automated Code Quality Measures

CISQ has developed Automated Quality Characteristic Measures to measure and manage the structural quality of IT application software. The automated measures for Security, Reliability, Performance Efficiency, and Maintainability are now OMG® approved standards making them global standards for use by IT organizations.

These measures were developed from coding rules covering some of the most serious violations of good architectural and coding practices that should be avoided and can be detected through static code analysis. Each measure counts the number of violations of the architectural and coding rules related to that quality characteristic, and then can be used in creating metrics for defect density, etc.

Security	Critical security violations in the source code drawn from the Top 25 security weaknesses in the Common Weakness Enumeration (CWE) repository
Reliability	Critical violations of availability, fault tolerance, and recoverability of software
Performance Efficiency	Critical violations of response time, as well as processor, memory, and utilization of other resources by the software
Maintainability	Critical violations of modularity, architectural compliance, reusability, analyzability, and changeability in software

About CAST CISQ Compliance Assessment

CAST offers CISQ compliant application assessment. Currently CAST supports 76 of the CISQ-86 with roadmaps to complete the coverage. The scoring mechanism CAST follows is consistent with ISO best practices, simple and easy to understand. Below are the four software characteristics against which CISQ and CAST detects the violations-

1. Security
2. Reliability
3. Performance efficiency
4. Maintainability

Assessment Results

Current Snapshot

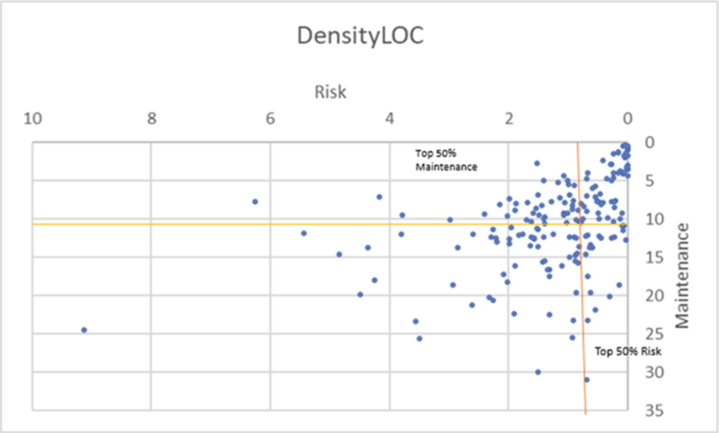
Application name	Gridlock
Version	9.0.1
Analysis date	October 13. 2020
Total violations	1229
Added violations	230
Removed violations	236
Δ from previous snapshots	-6

Assessment Summary

SOFTWARE CHARACTERISTICS	TOTAL	ADDED	FIXED	OPPS	%	APPMARQ RANK	APPMARQ QUARTILE
Security	83	26	41	778	99%	21/312	1st (93 %ile)
Reliability	322	54	68	707	98%	43/312	1st (86 %ile)
Performance Efficiency	442	80	69	796	96%	56/312	1st (82 %ile)
Maintainability	362	70	58	793	97%	37/312	1st (88 %ile)

TOTAL – Total number of violations in the application
ADDED – Number of violations added in last release
FIXED – Number of violations removed in last release
OPPS – Number of opportunities (No of artifacts, which has the potential to introduce risks)
% – Compliance percentage (1- Total Violations/ No of Opportunity)

Gridlock is in the top 15% of applications in its class and has improved in both aspects. This represents the operational stability (Reliability, Security and Performance Efficiency) and Maintainability, which is adjusted for Lines of Code



Details of violations

Security

RULE ID	RULE NAME	TOTAL	ADDED	FIXED	OPPS	%
ASCSM-CWE-022:	**Path Traversal Improper Input Neutralization**	10	6	7	756	99%
ASCSM-CWE-078:	**OS Command Injection Improper Input Neutralization**	12	2	10	609	99%
ASCSM-CWE-079:	**Cross-site Scripting Improper Input Neutralization**	7	6	9	547	99%
ASCSM-CWE-089:	**SQL Injection Improper Input Neutralization**	3	2	3	999	99%
ASCSM-CWE-99:	**Name or Reference Resolution Improper Input Neutralization**	5	3	4	803	96%
ASCSM-CWE-134:	**Format String Improper Input Neutralization**	24	6	6	902	96%
ASCSM-CWE-396:	**Declaration of Catch for Generic Exception**	22	1	2	876	95%
ASCSM-CWE-397:	**Declaration of Throws for Generic Exception**	0	0	0	917	100%
ASCSM-CWE-434:	**File Upload Improper Input Neutralization**	0	0	0	591	100%
ASCSM-CWE-456:	**Storable and Member Data Element Missing Initialization**	0	0	0	821	100%
ASCSM-CWE-772:	**Missing Release of Resource after Effective Lifetime**	0	0	0	514	100%
ASCSM-CWE-835:	**Loop with Unreachable Exit Condition (Infinite Loop)**	0	0	0	520	100%

Reliability

RULE ID	RULE NAME	TOTAL	ADDED	FIXED	OPP S	%
ASCRM-CWE-252-data:	Unchecked Return Parameter Value of named Callable and Method Control Element with Read, Write, and Manage Access to Data Resource	35	8	10	973	97%
ASCRM-CWE-252-resource:	Unchecked Return Parameter Value of named Callable and Method Control Element with Read, Write, and Manage Access to Platform Resource	30	8	8	880	95%
ASCRM-CWE-396:	Declaration of Catch for Generic Exception	48	4	10	563	93%
ASCRM-CWE-397:	Declaration of Throws for Generic Exception	23	6	4	684	94%
ASCRM-CWE-456:	Storable and Member Data Element Missing Initialization	46	7	1	702	93%
ASCRM-CWE-674:	Uncontrolled Recursion	28	10	7	734	97%
ASCRM-CWE-788:	Memory Location Access After End of Buffer		4	1	942	100%
ASCRM-RLB-01:	Empty Exception Block	39	1	7	712	95%
ASCRM-RLB-03:	Serializable Storable Data Element with non-Serializable Item Elements	26	3	10	795	96%
ASCRM-RLB-04:	Persistent Storable Data Element without Proper Comparison Control Element	47	3	10	946	95%
ASCRM-RLB-05:	Runtime Resource Management Control Element in a Component Built to Run on Application Servers	0	0	0	968	100%
ASCRM-RLB-08:	Named Callable and Method Control Elements with Variadic Parameter Element	0	0	0	929	100%
ASCRM-RLB-09:	Float Type Storable and Member Data Element Comparison with Equality Operator	0	0	0	661	100%
ASCRM-RLB-10:	Data Access Control Element from Outside Designated Data Manager Component	0	0	0	627	100%
ASCRM-RLB-11:	Named Callable and Method Control Element in Multi-Thread Context with non-Final Static Storable or Member Element	0	0	0	843	100%
ASCRM-RLB-12:	Singleton Class Instance Creation without Proper Lock Element Management	0	0	0	857	100%
ASCRM-RLB-13:	Inter-Module Dependency Cycles	0	0	0	906	100%
ASCRM-RLB-14:	Parent Class Element with References to Child Class Element	0	0	0	769	100%
ASCRM-RLB-18:	Storable and Member Data Element Initialization with Hard-Coded Network Resource Configuration Data	0	0	0	590	100%
ASCRM-RLB-19:	Synchronous Call Time-Out Absence	0	0	0	981	100%

Performance Efficiency

RULE ID	RULE NAME	TOTAL	ADDED	FIXED	OPPS	%
ASCPEM-PRF-02:	**Immutable Storable and Member Data Element Creation**	37	8	8	826	97%
ASCPEM-PRF-03:	**Static Member Data Element outside of a Singleton Class Element**	43	4	2	858	96%
ASCPEM-PRF-05:	**Data Resource Read Access Unsupported by Index Element**	35	8	4	631	96%
ASCPEM-PRF-06:	**Large Data Resource ColumnSet Excessive Number of Index Elements**	26	6	3	994	97%
ASCPEM-PRF-07:	**Large Data Resource ColumnSet with Index Element of Excessive Size**	24	6	8	554	94%
ASCPEM-PRF-08:	**Control Elements Requiring Significant Resource Element within Control Flow Loop Block**	35	7	2	906	95%
ASCPEM-PRF-09:	**Non-Stored SQL Callable Control Element with Excessive Number of Data Resource Access**	50	10	5	897	96%
ASCPEM-PRF-10:	**Non-SQL Named Callable and Method Control Element with Excessive Number of Data Resource Access**	34	1	10	825	97%
ASCPEM-PRF-11:	**Data Access Control Element from Outside Designated Data Manager Component**	27	8	3	581	94%
ASCPEM-PRF-12:	**Storable and Member Data Element Excessive Number of Aggregated Storable and Member Data Elements**	43	9	3	681	94%
ASCPEM-PRF-13:	**Data Resource Access not using Connection Pooling capability**	23	4	2	574	95%
ASCPEM-PRF-14:	**Storable and Member Data Element Memory Allocation Missing De-Allocation Control Element**	48	7	10	568	96%
ASCPEM-PRF-15:	**Storable and Member Data Element Reference Missing De-Referencing Control Element**	37	2	9	916	96%

Maintainability

RULE ID	RULE NAME	TOTAL	ADDED	FIXED	OPPS	%
ASCMM-MNT-03:	Storable and Member Data Element Initialization with Hard-Coded Literals	20	7	7	535	94%
ASCMM-MNT-05:	Loop Value Update within the Loop	20	4	5	553	94%
ASCMM-MNT-06:	Commented Code Element Excessive Volume	25	10	4	805	95%
ASCMM-MNT-07:	Inter-Module Dependency Cycles	39	10	7	620	97%
ASCMM-MNT-08:	Source Element Excessive Size	30	10	1	614	97%
ASCMM-MNT-09:	Horizontal Layer Excessive Number	49	5	6	836	96%
ASCMM-MNT-10:	Named Callable and Method Control Element Multi-Layer Span	42	3	2	675	93%
ASCMM-MNT-11:	Callable and Method Control Element Excessive Cyclomatic Complexity Value	44	7	5	845	94%
ASCMM-MNT-12:	Named Callable and Method Control Element with Layer-skipping Call	33	2	10	926	95%
ASCMM-MNT-13:	Callable and Method Control Element Excessive Number of Parameters	36	4	9	859	96%
ASCMM-MNT-14:	Callable and Method Control Element Excessive Number of Control Elements involving Data Element from Data Manager or File Resource	24	8	2	974	97%
ASCMM-MNT-15:	Public Member Element	0	0	0	578	100%
ASCMM-MNT-16:	Method Control Element Usage of Member Element from other Class Element	0	0	0	820	100%
ASCMM-MNT-17:	Class Element Excessive Inheritance Level	0	0	0	994	100%
ASCMM-MNT-18:	Class Element Excessive Number of Children	0	0	0	885	100%
ASCMM-MNT-19:	Named Callable and Method Control Element Excessive Similarity	0	0	0	509	100%
ASCMM-MNT-20:	Unreachable Named Callable or Method Control Element	0	0	0	958	100%

Location of Violations

Security

ASCSM-CWE-022: Path Traversal Improper Input Neutralization

CLASS PATH	LINE	STATUS
com.ie.atom.core.shelf.edit.userValidation.doPost	79	
com.ie.atom.core.rack.edit.userValidation.doPost	76	
com.ie.atom.core.exchange.edit.userValidation.doPost	56	
com.ie.atom.core.patchpanel.edit.userValidation.doPost	27	
com.ie.atom.core.FPP.edit.userValidation.doPost	39	Added
com.ie.atom.core.Card.edit.userValidation.doPost	86	Added
com.ie.atom.core.SubCard.edit.userValidation.doPost	72	Added
com.ie.atom.core.ModuleCard.edit.userValidation.doPost	59	Added
com.ie.atom.core.SOF.edit.userValidation.doPost	97	Added
com.ie.atom.core.DDF.edit.userValidation.doPost	106	Added

ASCSM-CWE-078: OS Command Injection Improper Input Neutralization

CLASS PATH	LINE	STATUS
com.ie.atom.admin.shelf.edit.edit.doPost	23	
com.ie.atom.admin.rack.edit.edit.doPost	54	
com.ie.atom.admin.exchange.edit.edit.doPost	78	
com.ie.atom.admin.patchpanel.edit.edit.doPost	114	
com.ie.atom.admin.FPP.edit.edit.doPost	21	
com.ie.atom.admin.Card.edit.edit.doPost	87	
com.ie.atom.admin.SubCard.edit.edit.doPost	67	
com.ie.atom.admin.ModuleCard.edit.edit.doPost	73	
com.ie.atom.admin.SOF.edit.edit.doPost	18	
com.ie.atom.admin.DDF.edit.edit.doPost	99	
com.ie.atom.admin.shelf.edit.edit.doPost	46	Added
com.ie.atom.admin.rack.edit.edit.doPost	76	Added

ASCSM-CWE-079: Cross-site Scripting Improper Input Neutralization

CLASS PATH	LINE	STATUS
Com.ie.atom.admin.patchpanel.home.doPost	32	
Com.ie.atom.admin.rack.home.doPost	21	Added
Com.ie.atom.admin.shelf.home.doPost	45	Added
Com.ie.atom.admin.shelf.home.doPost	56	Added
Com.ie.atom.admin.patchpanel.home.doPost	29	Added
Com.ie.atom.admin.rack.home.doPost	96	Added
Com.ie.atom.admin.shelf.home.doPost	57	Added

ASCSM-CWE-089: SQL Injection Improper Input Neutralization

CLASS PATH	LINE	STATUS
Com.ie.atom.admin.patchpanel.save.doPost	79	
Com.ie.atom.admin.rack.save.doPost	76	Added
Com.ie.atom.admin.shelf.save.doPost	56	Added

ASCSM-CWE-089: SQL Injection Improper Input Neutralization

CLASS PATH	LINE	STATUS
Com.ie.atom.admin.patchpanel.home.doPost	89	
Com.ie.atom.admin.rack.home.doPost	76	
Com.ie.atom.admin.shelf.home.doPost	34	Added
com.ie.atom.admin.shelf.edit.edit.doPost	98	Added
com.ie.atom.admin.rack.edit.edit.doPost	65	Added

This is a sample artifact. The details of other rules have been removed.

Appendix

About CAST AIP

CAST AIP connects into all major SCM systems. CAST AIP can also be configured to analyze application source code in whatever format it is maintained in the organization. Source code is then processed and stored in the CAST Knowledge Base as metadata, which forms the basis for the analysis and information provided by CAST AIP. CAST assesses the **entire** application as a single entity, the only way to verify what is actually deployed into production —this can include legacy components, packaged app customizations, frameworks and all modern distributed technology environments. Data from third party code analyzers can be integrated into the CAST Knowledge Base and displayed in AIP dashboards. CAST AIP integrates seamlessly into DevOps or DevSecOps automation chains to allow for ongoing automation. CAST AIP is deployed within many of the largest software factories in the world.

About CISQ Software Characteristics

Security

Security assesses the degree to which an application protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization (ISO 25010). Security measures the risk of potential security breaches due to poor coding and architectural practices. Security problems have been studied extensively by the Software Assurance community and have been codified in the Common Weakness Enumeration (CWE) at cwe.mitre.org.

The CISQ Automated Source Code Security Measure draws from the CWE/SANS Institute Top 25 Most Dangerous Software Errors and identifies the most widespread and frequently exploited security weaknesses in software. Twenty-two of these weaknesses are detectable through analyzing the source code and form the basis of the CISQ measure. These 22 weaknesses constitute the most frequent ways unauthorized parties breach a system. Thus, the CISQ measure is a good predictor of how easily an application can suffer unauthorized penetration that results in stolen information, altered records, or other forms of malicious behavior.

Reliability

Reliability measures the risk of potential application failures and the stability of an application when confronted with unexpected conditions. According to ISO/IEC/IEEE 24765, Reliability is the degree to which a system, product, or component performs specified functions under specified conditions for a specified period of time. The reason for checking and monitoring Reliability is to prevent or at least reduce application downtime, outages, data corruption, and errors that directly affect users.

The CISQ Automated Source Code Reliability Measure is composed from 29 critical violations of architectural and coding practice that affect the availability, fault tolerance, recoverability, and data integrity of an application. The CISQ Reliability measure produces a quality score based on the count of violations discovered in the software and can be turned into a density measure when divided by the size of the software.

Performance Efficiency

Performance Efficiency assesses characteristics that affect an application’s response behavior and use of resources under stated conditions (ISO/IEC 25010). Performance Efficiency affects customer satisfaction, workforce productivity, application scalability, response-time degradation, and inefficient use of processing or storage resources. The Performance Efficiency of an application lies in each individual component ‘s performance, as well as in the effect of each component on the behavior of the chain of components comprising a transaction in which it participates.

The CISQ Automated Source Code Performance Efficiency Measure is composed from 15 critical violations of response time behavior, processor use, and memory use of an application. A quality score is produced based on the count of violations discovered in the source code and can be used as a density metric when divided by software size.

Maintainability

Maintainability represents the degree of effectiveness and efficiency with which a product or system can be modified by the intended maintainers (ISO 25010). Maintainability incorporates such concepts as changeability, modularity, understandability, testability, and reusability. Maintainability is responding rapidly to market conditions and keeping IT costs under control. The Maintainability of an application is a combination of compliance with good coding practices, the homogeneity with which coding rules are applied across an application, and compliance with architectural rules.

The CISQ Automated Source Code Maintainability Measure is composed from 20 critical violations that reduce the maintainability of a software application. A quality score is produced based on the count of violations discovered in the software that can be used as a density metric when divided by software size.

CAST Coverage of CISQ Rules

Security

SR NO	RULE ID	TOTAL VIOLATIONS	CAST SUPPORT
1	ASCSM-CWE-022:	Path Traversal Improper Input Neutralization	Yes
2	ASCSM-CWE-078:	OS Command Injection Improper Input Neutralization	Yes
3	ASCSM-CWE-079:	Cross-site Scripting Improper Input Neutralization	Yes
4	ASCSM-CWE-089:	SQL Injection Improper Input Neutralization	Yes
5	ASCSM-CWE-99:	Name or Reference Resolution Improper Input Neutralization	Yes
6	ASCSM-CWE-120:	Buffer Copy without Checking Size of Input	No
7	ASCSM-CWE-129:	Array Index Improper Input Neutralization	No
8	ASCSM-CWE-134:	Format String Improper Input Neutralization	Yes
9	ASCSM-CWE-252-resource:	Unchecked Return Parameter Value of named Callable and Method Control Element with Read, Write, and Manage Access to Platform Resource	No
10	ASCSM-CWE-327:	Broken or Risky Cryptographic Algorithm Usage	No
11	ASCSM-CWE-396:	Declaration of Catch for Generic Exception	Yes
12	ASCSM-CWE-397:	Declaration of Throws for Generic Exception	Yes
13	ASCSM-CWE-434:	File Upload Improper Input Neutralization	Yes

SR NO	RULE ID	TOTAL VIOLATIONS	CAST SUPPORT
14	ASCSM-CWE-456:	Storable and Member Data Element Missing Initialization	Yes
15	ASCSM-CWE-606:	Unchecked Input for Loop Condition	No
16	ASCSM-CWE-667:	Shared Resource Improper Locking	No
17	ASCSM-CWE-672:	Expired or Released Resource Usage	No
18	ASCSM-CWE-681:	Numeric Types Incorrect Conversion	No
19	ASCSM-CWE-772:	Missing Release of Resource after Effective Lifetime	Yes
20	ASCSM-CWE-789:	Uncontrolled Memory Allocation	No
21	ASCSM-CWE-798:	Hard-Coded Credentials Usage for Remote Authentication	No
22	ASCSM-CWE-835:	Loop with Unreachable Exit Condition (Infinite Loop)	Yes

Reliability

SR NO	RULE ID	TOTAL VIOLATIONS	CAST SUPPORT
1	ASCRM-CWE-120:	Buffer Copy without Checking Size of Input	No
2	ASCRM-CWE-252-data:	Unchecked Return Parameter Value of named Callable and Method Control Element with Read, Write, and Manage Access to Data Resource	Yes
3	ASCRM-CWE-252-resource:	Unchecked Return Parameter Value of named Callable and Method Control Element with Read, Write, and Manage Access to Platform Resource	Yes
4	ASCRM-CWE-396:	Declaration of Catch for Generic Exception	Yes
5	ASCRM-CWE-397:	Declaration of Throws for Generic Exception	Yes
6	ASCRM-CWE-456:	Storable and Member Data Element Missing Initialization	Yes
7	ASCRM-CWE-674:	Uncontrolled Recursion	Yes
8	ASCRM-CWE-704:	Incorrect Type Conversion or Cast	No
9	ASCRM-CWE-772:	Missing Release of Resource after Effective Lifetime	Yes
10	ASCRM-CWE-788:	Memory Location Access After End of Buffer	Yes
11	ASCRM-RLB-01:	Empty Exception Block	Yes
12	ASCRM-RLB-02:	Serializable Storable Data Element without Serialization Control Element	No
13	ASCRM-RLB-03:	Serializable Storable Data Element with non-Serializable Item Elements	Yes
14	ASCRM-RLB-04:	Persistent Storable Data Element without Proper Comparison Control Element	Yes
15	ASCRM-RLB-05:	Runtime Resource Management Control Element in a Component Built to Run on Application Servers	Yes
16	ASCRM-RLB-06:	Storable or Member Data Element containing Pointer Item Element without Proper Copy Control Element	N/A (Not applicable in the Java context, where there is no pointer)
17	ASCRM-RLB-07:	Class Instance Self Destruction Control Element	N/A (Not applicable in the Java context, where there is no explicit destruction of objects)

SR NO	RULE ID	TOTAL VIOLATIONS	CAST SUPPORT
18	ASCRM-RLB-08:	Named Callable and Method Control Elements with Variadic Parameter Element	Yes
19	ASCRM-RLB-09:	Float Type Storable and Member Data Element Comparison with Equality Operator	Yes
20	ASCRM-RLB-10:	Data Access Control Element from Outside Designated Data Manager Component	Yes
21	ASCRM-RLB-11:	Named Callable and Method Control Element in Multi-Thread Context with non-Final Static Storable or Member Element	Yes
22	ASCRM-RLB-12:	Singleton Class Instance Creation without Proper Lock Element Management	Yes
23	ASCRM-RLB-13:	Inter-Module Dependency Cycles	Yes
24	ASCRM-RLB-14:	Parent Class Element with References to Child Class Element	Yes
25	ASCRM-RLB-15:	Class Element with Virtual Method Element without Virtual Destructor	N/A (Not applicable in the Java context, where there is no explicit destruction of objects)
26	ASCRM-RLB-16:	Parent Class Element without Virtual Destructor Method Element	N/A (Not applicable in the Java context, where there is no explicit destruction of objects)
27	ASCRM-RLB-17:	Child Class Element without Virtual Destructor unlike its Parent Class Element	N/A (Not applicable in the Java context, where there is no explicit destruction of objects)
28	ASCRM-RLB-18:	Storable and Member Data Element Initialization with Hard-Coded Network Resource Configuration Data	Yes
29	ASCRM-RLB-19:	Synchronous Call Time-Out Absence	Yes

Efficiency

SR NO	RULE ID	TOTAL VIOLATIONS	CAST SUPPORT
1	ASCPEM-PRF-01	Static Block Element containing Class Instance Creation Control Element	No
2	ASCPEM-PRF-02:	Immutable Storable and Member Data Element Creation	Yes
3	ASCPEM-PRF-03:	Static Member Data Element outside of a Singleton Class Element	Yes
4	ASCPEM-PRF-04:	Data Resource Read and Write Access Excessive Complexity	No
5	ASCPEM-PRF-05:	Data Resource Read Access Unsupported by Index Element	Yes
6	ASCPEM-PRF-06:	Large Data Resource Column Set Excessive Number of Index Elements	Yes
7	ASCPEM-PRF-07:	Large Data Resource Column Set with Index Element of Excessive Size	Yes
8	ASCPEM-PRF-08:	Control Elements Requiring Significant Resource Element within Control Flow Loop Block	Yes
9	ASCPEM-PRF-09:	Non-Stored SQL Callable Control Element with Excessive Number of Data Resource Access	Yes
10	ASCPEM-PRF-10:	Non-SQL Named Callable and Method Control Element with Excessive Number of Data Resource Access	Yes
11	ASCPEM-PRF-11:	Data Access Control Element from Outside Designated Data Manager Component	Yes
12	ASCPEM-PRF-12:	Storable and Member Data Element Excessive Number of Aggregated Storable and Member Data Elements	Yes
13	ASCPEM-PRF-13:	Data Resource Access not using Connection Pooling capability	N/A (Not applicable in the Java context, where memory is fully managed)
14	ASCPEM-PRF-14:	Storable and Member Data Element Memory Allocation Missing De-Allocation Control Element	Yes
15	ASCPEM-PRF-15:	Storable and Member Data Element Reference Missing De-Referencing Control Element	Yes

Maintainability

SR NO	RULE ID	TOTAL VIOLATIONS	CAST SUPPORT
1	ASCMM-MNT-01:	Control Flow Transfer Control Element outside Switch Block	No
2	ASCMM-MNT-02:	Class Element Excessive Inheritance of Class Elements with Concrete Implementation	No
3	ASCMM-MNT-03:	Storable and Member Data Element Initialization with Hard-Coded Literals	Yes
4	ASCMM-MNT-04:	Callable and Method Control Element Number of Outward Calls	No
5	ASCMM-MNT-05:	Loop Value Update within the Loop	Yes
6	ASCMM-MNT-06:	Commented Code Element Excessive Volume	Yes
7	ASCMM-MNT-07:	Inter-Module Dependency Cycles	Yes
8	ASCMM-MNT-08:	Source Element Excessive Size	Yes
9	ASCMM-MNT-09:	Horizontal Layer Excessive Number	Yes
10	ASCMM-MNT-10:	Named Callable and Method Control Element Multi-Layer Span	Yes
11	ASCMM-MNT-11:	Callable and Method Control Element Excessive Cyclometric Complexity Value	Yes
12	ASCMM-MNT-12:	Named Callable and Method Control Element with Layer-skipping Call	Yes
13	ASCMM-MNT-13:	Callable and Method Control Element Excessive Number of Parameters	Yes
14	ASCMM-MNT-14:	Callable and Method Control Element Excessive Number of Control Elements involving Data Element from Data Manager or File Resource	Yes
15	ASCMM-MNT-15:	Public Member Element	Yes
16	ASCMM-MNT-16:	Method Control Element Usage of Member Element from other Class Element	Yes
17	ASCMM-MNT-17:	Class Element Excessive Inheritance Level	Yes
18	ASCMM-MNT-18:	Class Element Excessive Number of Children	Yes
19	ASCMM-MNT-19:	Named Callable and Method Control Element Excessive Similarity	Yes

SR NO	RULE ID	TOTAL VIOLATIONS	CAST SUPPORT
20	ASCMM-MNT-20:	Unreachable Named Callable or Method Control Element	Yes