《MISRA-C 合规检验工具对比研究》概要

TERA 实验室（TERA-Labs）独立研究

(安特卫普·卡瑞尔格若特应用科学大学研究部)

（Karel de Grote University College, Antwerp）

2012 年 5 月 9 日，TERA 实验室（比利时·安特卫普·卡瑞尔格若特应用科学大学的研究部）发布了《MISRA-C 合规检验工具对比研究》报告的最终版本。[参考 1]

本白皮书由两部分组成：

- **第一部分** 对 TERA 实验室所做的 80 页的研究进行了一个全面、详细的概述。该概述由 PRQA 生成，其中包含对 TERA 实验室所作报告的关键的直接引用。TERA 实验室做研究报告的原作者已经确认，认为该概述清楚、准确地概括了原报告。

- **第二部分** PRQA 根据 TERA 实验室所作的报告提出了自己的观点和意见。

关键词：MISRA，静态代码分析，编码规范，合规

第一部分： 报告概述

本部分包含对 TERA 实验室 80 页的详细原报告所作的概述【参考 1】。

1.1 引言

TERA 实验室是卡瑞尔格若特应用科学大学（Karel de Grote University College）的研究部，该大学坐落在比利时的安特卫普。TERA 实验室所做的研究主要集中在各个应用程序/行业中的嵌入式系统方面，研究的范围十分广泛，大至汽车工业，小至分布计算程序。此次专门对 MISRA-C 合规检验工具进行的独立研究在 IWT（agentschap voor Innovatie door Wetenschap en Technologie, www.iwt.be）资助下完成，该研究持续了 20 个月。该研究报告的最初版本发布于 2010 年 10 月 6 日，最终版本发布于 2012 年 5 月 9 日。

1.2 研究范围和目标

该研究的目标是评估软件工具在实施 MISRA-C:2004 编码规则方面的有效性。评估方法是 TERA 实验室的工程师用 C 编程语言写一系列测试用例（“试样”），并用各个工具中对这些测试用例进行测试，而每个测试用例中都包含故意违反 MISRA-C:2004 编码规则的代码。研究团队通过各个工具识别这些违规代码的情况，来评估每个工具发现违规代码的能力。

1.3 参与者

TERA 实验室选取了以下静态分析/代码分析工具进行研究。这些工具是通过因特网搜索选出的，它们都明确提到支持 MISRA-C 规则的检测：

1. Development Assistant for C (DAC) - RistanCASE
2. IAR embedded workbench
3. Klocwork Insight
4. LDRA Testbed
5. Parasoft C++test
6. QA·C – Programming Research/PRQA
7. PC-Lint - Gimpel
8. (Prevent - Coverity) *
9. Raincode **

（* TERA 实验室在发现 Coverity 的工具只支持 MISRA-C 规则的最小子集之后，立刻放弃了继续对其进行研究，所以报告的结果部分并没有 Coverity 的相关数据。**之后将 Raincode 加入到研究项目中，所以结果中包含了 Raincode 的数据。）

另外需要注意的是，根据法律要求，在 TERA 实验室所作的最终报告中，工具供应商都是匿名的，这些供应商被称为 Company 01 到 09。上面所列出的工具的顺序和图表以及表格中的列的顺序并没有关联。但是，在这份总结性的白皮书中，我们可以确定 PRQA (QA•C)是 Company04。

1.4 研究标准

在研究中“软性”和“硬性”评估标准均被考虑在内，主要有以下标准：

软性标准：

1. 方便在现有开发环境中集成。
2. 可用性：方便用户使用某些特定的工具。
3. 扩展性：用户是否可以很方便地利用附加功能对项目进行扩展？

4. 违规信息的质量。
5. 命令行功能（自动化）。
6. 附加功能，如：度量？

硬性标准：

7. 正确性：工具报告的违规信息是否正确（正确报警）？
8. 完整性：工具是否能报告代码中的所有违规信息（没有漏报）？

要对所有的 MISRA-C 规则进行测试是不现实的，所以，该研究集中对 11 个“重要”和“典型”的规则子集进行了测试。这些规则是工业伙伴的专家组选出来的。

这些规则分为 3 个小组：

1. 毁灭性：如果违反这些规则，可能带来毁灭性影响。
2. 可维护性：这些规则可降低在修改代码时向项目中引进错误的可能性。
3. 可移植性：提高可移植性

本次研究对以下 11 条规则中，被划分到“毁灭性”和“可维护性”分组里的规则进行了评估：

MISRA-C 规则	规则描述
2.3	字符序列/*不可以用在注释中。
8.12	如果外部链接中公布了一个数组， 那么应该明确指出该数组的大小或者通过初始化暗中定义其大小。
9.1	在使用自动变量之前，要对其进行赋值操作。
11.1	函数的指针不可以和整数类以外的任何类型进行相互转换。
12.4	0≤移位算子的右手操作数≤左手操作数的基础类型的宽度（以比特单位）-1。
14.7	一个函数只能在函数末尾处有一个出口点。
15.2	无条件中断语句应该终止所有非空 switch 子句。
15.3	Switch 语句中的最后一个子句应为默认子句。
16.6	传递给函数的实参的数量与形参的数量一致。
17.6	有些对象在带有自动存储区的对象不复存在之后，依然可以继续存在。带有自动存储区的对象的地址不可以分配给这些对象。
19.10	在函数宏定义中，每个形参的例子都必须写在括号中，除非这些例子是被当作操作数# 或 ##。

1.5 结果 – 软性标准

TERA 实验室的报告包含 9 个表，我们将这些表的内容总结在下面这一个表中：

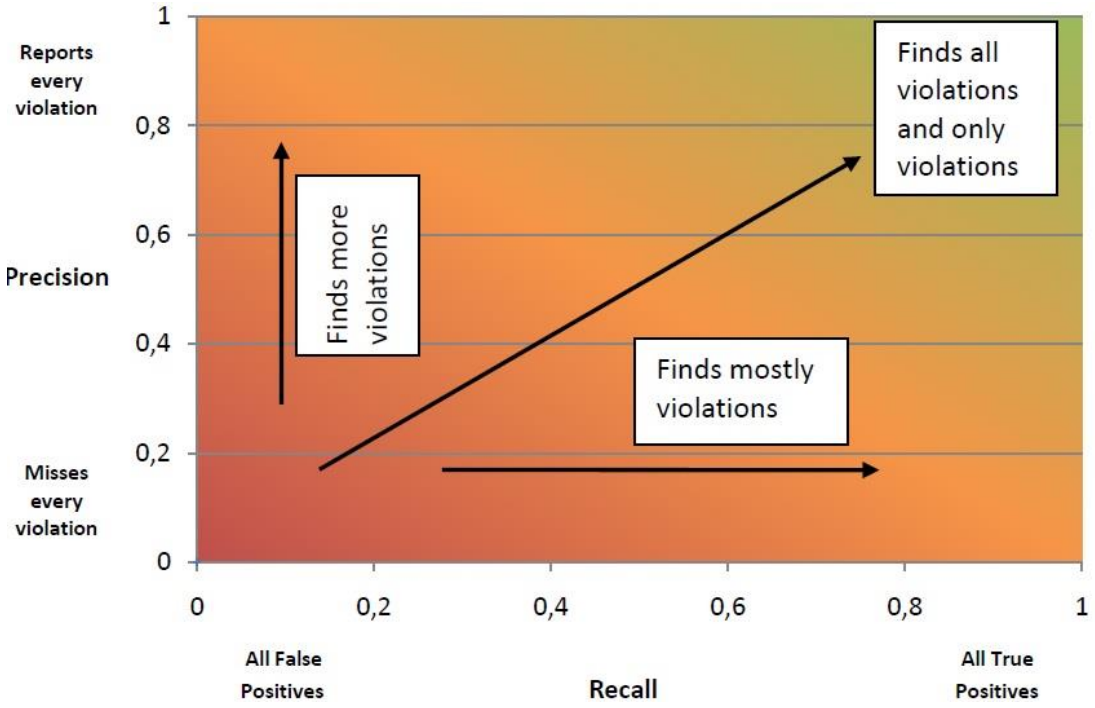
标准	Comp 01	Comp 02	Comp 03	Comp 04 (PRQA)	Comp 05	Comp 06	Comp 08	Comp 09	最大
在命令行中切换规则	0	2	2	3	3	0	0	0	**3**
在图形化用户界面切换规则	3	3	3	3	n/a	1	3	3	**3**
分析项目并排除文件	3	3	4	4	3	3	4	4	**4**
错误信息和警报的质量	3	3	5	6	3	6	3	4	**6**
集成	1	1	2	1	2	1	1	1	**3**
支持的操作系统	1	1	1	2	1	2	2	2	**2**
自动化	1	1	0	1	1	1	1	0	**1**
附加功能	3	1	3	2	1	3	4	1	**5**
技术支持	0	3	2	3	2	3	3	3	**3**
总分	**15**	**18**	**22**	**25**	**16**	**20**	**21**	**18**	**30**

（注意：在完整的报告中，这 9 个表对几个问题做了“是” 或“否”的回答，这些问题可按等级划分为 1-3 星。在上面这个浓缩/简化的表中，“是”可得 1 分，一颗星也可得 1 分。最右边的“最大”列表示的是每个分类所能得到的最大分数。）

1.6 结果 – 硬性标准

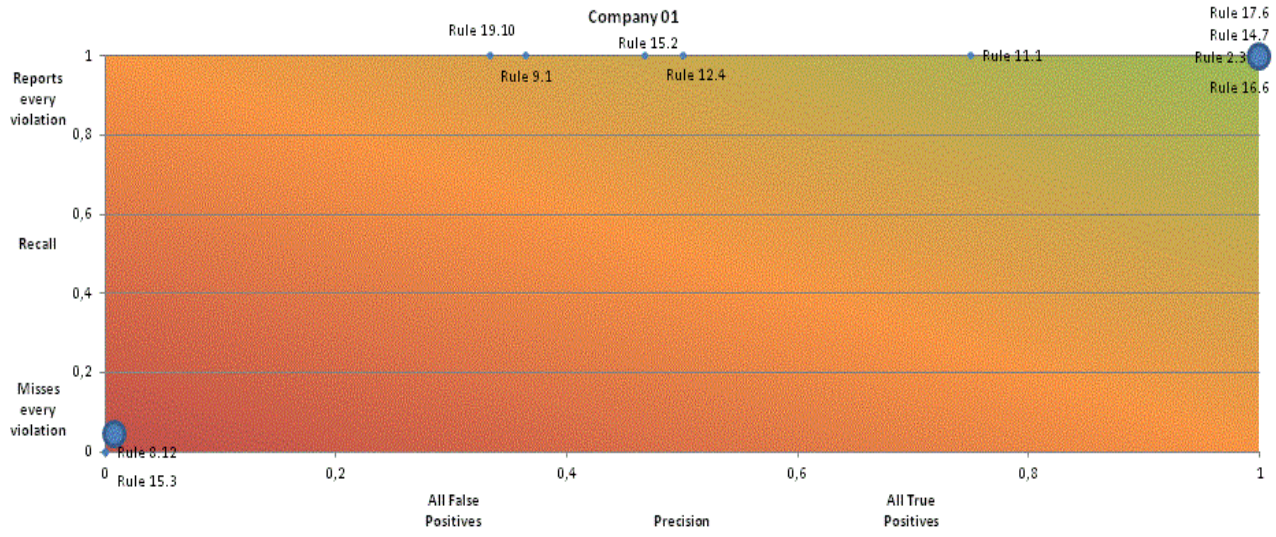
TERA 实验室研究的核心显示了硬性标准评估的结果，尤其是每个工具在发现违反那 11 个 MIRSA-C 规则的代码方面的有效性。这种有效性主要通过每个工具在以下两个方面的能力来衡量：1）发现并报告代码中所有真正违反规范的地方； 2）避免误报（干扰），如：对没有真正违规的地方进行违规报告。

结果显示在一系列的“精确度”-“检索率”的图表中。Y-轴（检索率）表示工具在发现真正违规信息方面的有效性。X-轴（精确度）表示工具的干扰性和产生误报的可能性有多大。

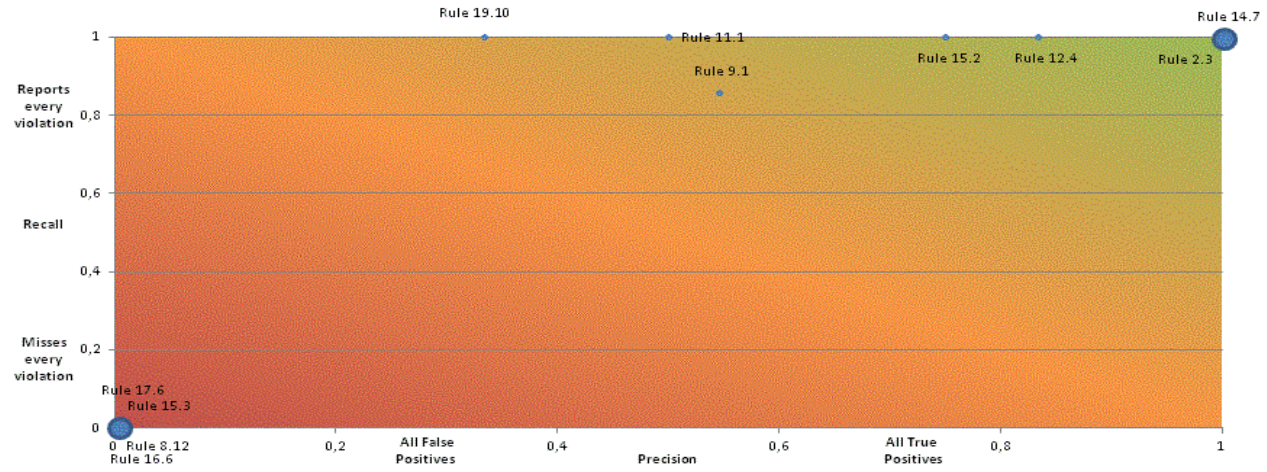


性能最好的工具会有数据点分布在图表的**右上角**，表示该工具能够成功找到违规代码，并且不会产生任何干扰/误报信息。数据点越靠左，表示工具的干扰性越强（产生的误报越多）；数据点越靠下表示工具发现违规代码的能力越弱。所以数据点分布在**左下角**的工具的性能最差，说明该工具无法发现违规代码，只会产生干扰/误报信息。

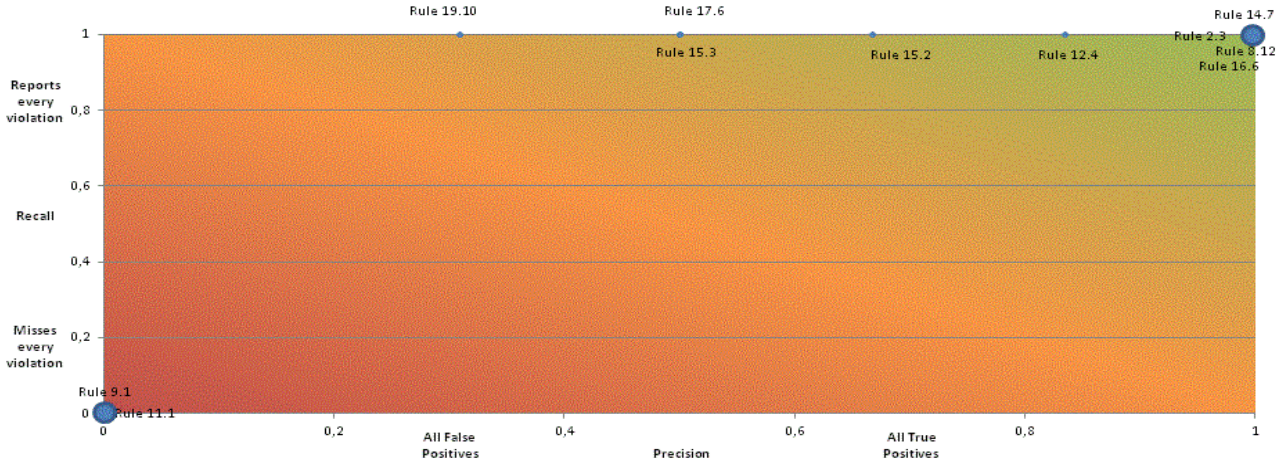
下面的图表是 TERA 实验室报告在附录 L 到 S 中内容，显示了 8 套工具的性能（除了 Coverity）：



Company 02

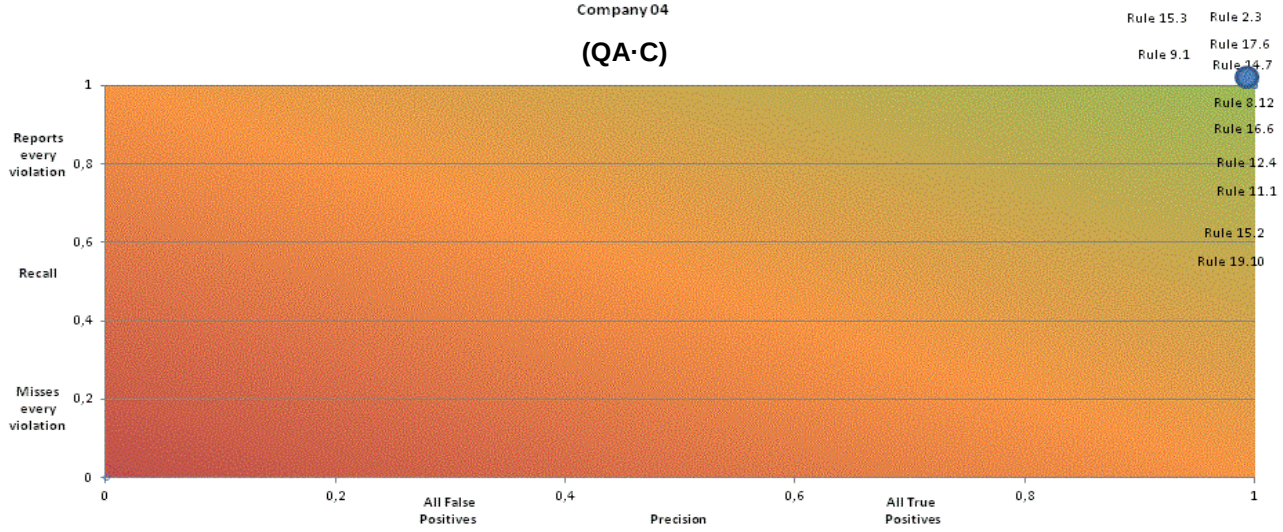


Company 03

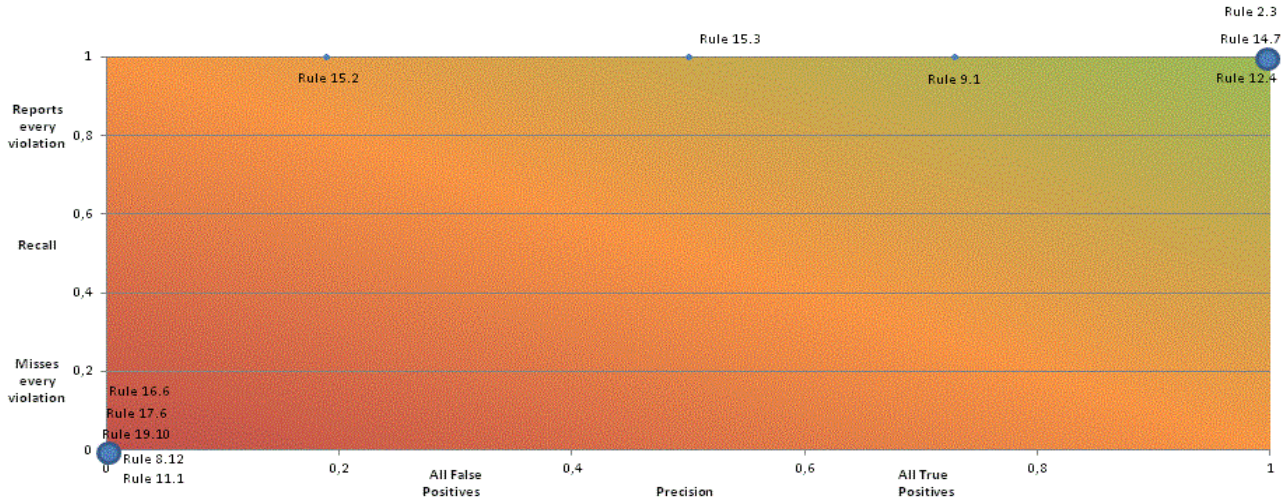


Company 04

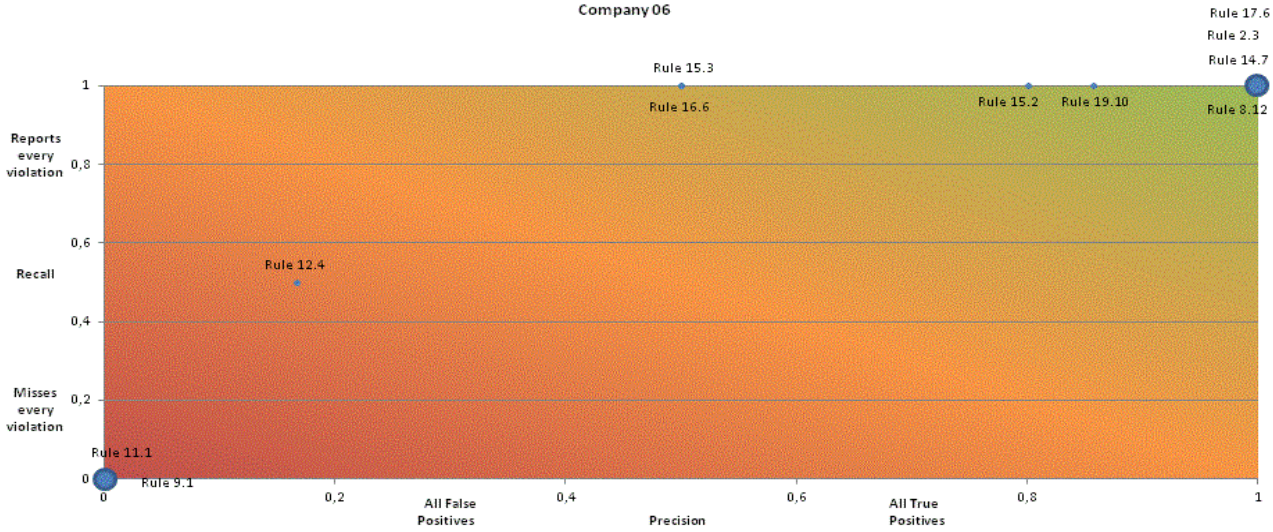
(QA·C)



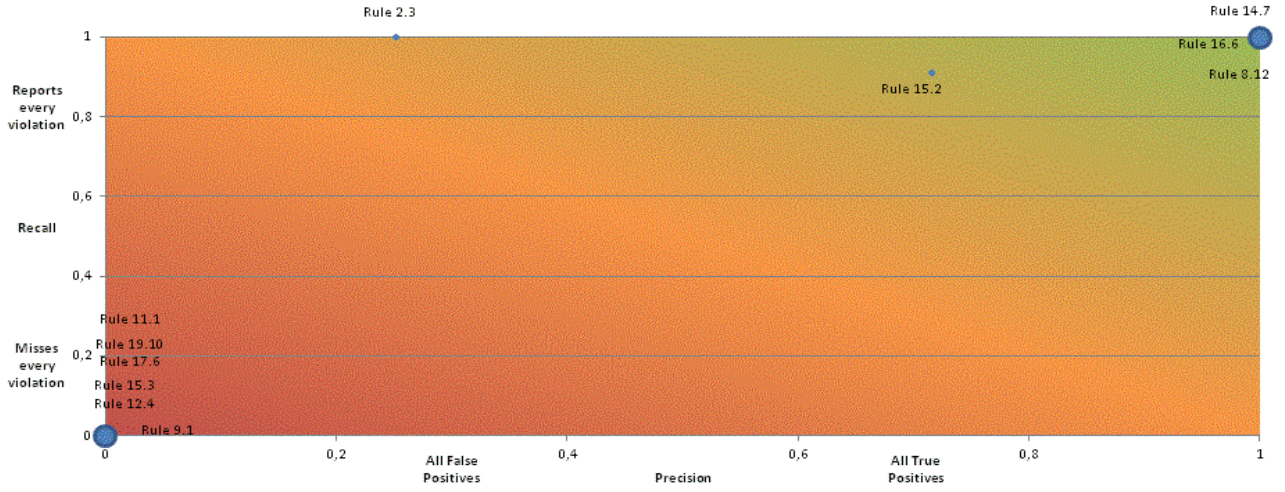
Company 05

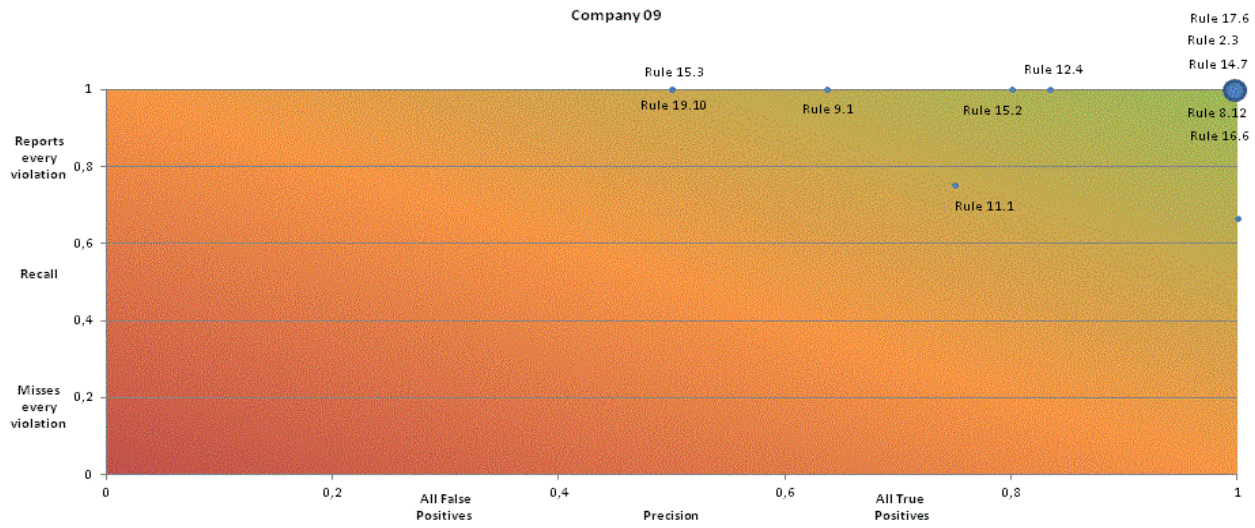


Company 06



Company 08





1.7 TERA 实验室研究总结

TERA 实验室所做的研究的简报只能提供分析和结果，绝对不能明确推荐哪个工具“最好”。

TERA 实验室报告的结论可概括为以下 6 点：

1. 工具间的差异很大
   - 图形化用户界面、命令行、安装程序、许可方案等等都存在很大差异。
   - 各个工具在违规规则方面也有很大差异。
2. MISRA 的规则很多，利用任何工具都会产生无数的违规信息。
3. 团队预计可能会发现大量的误报信息。但意料之外的是，团队实际上发现了更多的漏报的情况，如：有几套工具竟不能发现很明显的违规代码。
4. 通常情况下，工具越贵，测试的结果越好，比如：在“精确度-检索率”图表上很多规则取得的数据点都分布在*右上角*。
5. 大多数工具都没有获取完整注释源（源代码+违规信息+行号）并将其导出成文件的选项。这就使该工具无法与现有的软件开发步骤集成。
6. 所有的工具都有一个严重的问题（有些工具这方面的问题比其它工具更严重），即：如果不能对代码进行完整解析，工具就无法给出代码违规结果。这主要是大型现有数据库所面临的问题。

SECTION 2： PRQA 的观点和评论

文档的本小节将对 PRQA 的主要观点和评论进行概述。这些评论也反映了 TERA 报告的作者在几次额外的直接讨论中所表达的观点。

综述

我们强烈认为该研究是对 MISRA-C 合规检验工具进行的一次真实的、可靠的、独立的评估。（上面已经提到，该报告中也有几处不太准确的地方，我们可能会对有些解释存在一点疑问。）

TERA 实验室团队说：“报告中（根据营销材料的描述）选用的所有工具性能都是相当的，都可以进行全面的 MIRSA-C 合规检查，但是实际上，它们的性能差别很大。”

TERA 实验室团队花了很多时间和精力进行研究分析，从而得出客观的结论——超过 1 年。没有任何商业公司会在工具选择上给予这么大的投资。

测试用例

所选的 11 条规则确实是一个很好的子集——包含了 MISRA-C 编码规范的重要规则和典型。

TERA 实验室创建的测试用例中的代码违规行为比较简单、直白，并不是复杂的代码，也没有复杂逻辑或不常见的边界情况。我们都知道，如果工具在分析基本的编码违规行为时，作用不大，那么它在分析更大的复杂代码库时，就会更糟糕了。

结果- 硬性标准的图表

既能发现真正的违规代码，又能减少干扰/漏报情况，无疑是对工具的性能和效率进行评估的关键指标，而 TERA 实验室生成的图示则为每个工具的性能提供了极好的总结。

QA•C 的认可度这么高，我们感到非常高兴。它无疑是所有被测工具中性能最好的，因为它发现了代码中关于所选的 11 条编码规范的所有违规行为，而且没有产生任何误报/干扰信息。

我们可以做出以下结论：

- X-轴（精确度）表示工具产生干扰信息和生成误报的可能性。数据点在图表中的分布越靠左，意味着工具生成的误报信息越多。误报产生的最明显的影响就是开发人员需要花额外的时间/费用来分析、发现、排除这些误报信息。如果只是用简单的小型代码对少数规则进行评估，也许误报并不会成为一个严重的问题。但是，如果测试的规则很多，代码很大，那么干扰性大的工具所产生的额外费用和时间就会很明显。而且，我们也应该知道，工具产生太多的误报信息最终会损坏工具的信誉，因而就很少有开发团队会使用这套工具，这是非常现实的危害。

- Y-轴（检索率）表示工具在发现真正的违规代码方面的有效性。数据点在图表中的分布越靠下，表示工具发现真正违规代码的能力越差。漏报产生的影响可能比误报更严重。我们认为会产生太对漏报的工具“不适合”做编码规范的合规检查。我们应该注意到，TERA 实验室所作的报告中有一条关键的结论就是：他们没有想到会遇到那么多的漏报情况。**PRQA 能够发现其它工具没有发现的重要的、实质性的缺陷。**

推断结果

我们推断，如果这个研究覆盖了所有的（~133）MISRA-C 静态执行规则，那么每个工具的执行情况应该还是差不多，所得到的结果应该和检测所选的 11 条具有代表性的 MISRA-C 规则所得到的结果一致。TERA 实验室团队也赞同这一推断。实际上，我们估计工具供应商的工具在 MISRA-C++合规检测方面的性能可能也差不多。

投资回报率(Return on Investment)

我们注意到，TERA 实验室的研究主要集中在工具的技术性能方面，而并没有评估使用各个工具时，开发人员在发现和修复违规代码方面所花费的时间/费用。不过很明显，干扰信息性强的工具，需要开发人员花费更多的时间/费用去排除误报信息。另外，发现漏报信息同样需要花额外的时间/费用，而且还需要利用其它测试方法来找出那些漏报信息。遗憾的是，每个工具的成本效益分析和投资回报率并不在 TERA 实验室本次研究的范围之内。

过程

在与 TERA 实验室人员进行讨论过程中，我们都发现如果开发团队事前已经安排了适当的软件开发流程，那么他们会从 MISRA 和 MISRA 合规工具中受益更多。

编码规范

值得注意的是，虽然很多开发团队都努力让他们的代码完全符合 MISRA 的要求，但是有些团队只是为了提高开发代码的坚固性，而遵循 MISRA（或其它编码规范）的一些关键规则，比如：聚焦实质性的/严重的缺陷（如：不明确的行为），加强代码评审。他们的目标不是实现“符合 MISRA 标准的代码”而是使用 MISRA 来提高代码的坚固性。

一个非常有效又常见的方法就是，先选用 MISRA 中的一些关键规则作为子集（如：TERA 实验室做研究报告时，一开始只选用了其中 11 条规则），然后在开发团队逐渐了解和认可该编码规范之后，再采纳更多的规则要求。（使用 PRQA 工具，可以轻松选择 MISRA 规则子集，并可根据公司的预定规则对规则集进行补充。）

无论规则源于内部还是外部标准，都要了解静态分析工具在进行高效的自动化合规测试方面的作用。

建立在坚实的基础之上

PRQA 工具强大的性能反映了 PRQA 和 MISRA 之间长达 20 多年的合作关系。MISRA-C 和 MISRA-C++纲要的主要内容都从我们自己的编码规范中衍生而来的，而且我们的技术专家一直都是编写 MISRA 标准（包括新的 MISRA-C3 标准）的工作团队中的主要成员。我们的静态分析工具正是由这些语言专家设计和开发的。

总结

TERA 实验室的研究，是针对 MISRA-C 合规检验工具进行的一次真实的、可靠的、独立的评估。让我们感到高兴的是：QA•C 的认可度这么高，而且是所有被测工具中效果最好的——它能够发现其它工具无法发现的重要的、实质性的缺陷，而且几乎不会产生误报信息。我们想提醒各个公司在选择工具时，要意识到性能比较差的工具产生漏报、误报、干扰信息时将带来的影响（成本、时间、质量、投资回报率）。

参考：参考 1：安特卫普•卡瑞尔格若特应用科学大学（Karel de Grote University College, Antwerp）的 Marijn Temmerman 博士提供了独立于 TERA 实验室的研究项目——“MISRA 合规检查工具的对比”中获得的调查结果。
http://www.programmingresearch.com/hidden/tera-labs/

创提信息科技（上海）有限公司
www.trinitytec.com.cn

上海市浦东新区张江高科张衡路
200 号 3 幢 3207E 室，201204
T：21 - 3126 8126
F：21 - 5132 8526
E：info@trinitytec.net