



CHANGEMINER WHITE PAPER

应用程序的开发/维护效率提高 30~75% !



Copyright © 2015 GTOne Corp. All Rights Reserved.

Copyright in this document is vested in GTOne Corp. The contents of the document (wholly or in part) must not be reproduced, distributed used or disclosed without the prior written permission of GTOne Corp.

概要

当今大部分企业的软件无法充分满足快速持续变化的业务目标，即便是承担着高费用与高风险。软件开发团队总被期望在相同或更少资源的情况下，交付更多的软件。管理人员、开发人员、分析设计人员、质量管理人在处理大量日常业务时有必要提高他们的作业效率。

应用程序自动分析工具可以使IT专家对应用进行更深入的了解，从而帮助解决与应用相关的广泛的问题点。Forrester Research公司提出“应用程序自动分析工具可提高30%生产率”^[1]并根据GTONE针对各客户案例的分析，部分客户报告称提高到75%的生产率。

本白皮书将介绍应用程序相关的问题点以及解决方案应用程序自动分析工具。

[1] “Attack Weak Application Maintenance Processes That Stifle IT Productivity”, Phil Murphy, Forrester Research, 2007. 2.

应用程序问题 - 高费用低生产率

应用程序快速持续得变更带来的风险

IT服务的信赖性成为了当今企业日常业务处理中的一个必要的因素，但无法避免由频繁的业务变化导致应用程序也要发生大量的变更。失误的应用程序变更可能会导致IT系统的故障，进而增加造成商务损失的危险性。

开发人员在没有进行充分的影响分析或者对整体关联性了解不深的情况下，修改源代码或者数据库必然会增加发生系统故障的概率。

频繁的应用程序变更会给应用程序生命周期中的编译、发布及测试阶段带来影响。在没有确保应用程序可视性的情况下很难掌握正确的测试的范围，并会给编译、发布及测试人员带来负担。

日趋增加的复杂度和规模带来的应用程序的质量、规范问题

数十年来IT组织运维着复杂度无法想象的应用程序。应用程序的内部逻辑使用以各不相同的数据库与COBOL, 4GL, Java, .net等多种语言构成的组件。

并且应用程序的规模也在持续地快速增加。在此关键时刻，应用程序运维团队对应用程序可视性的掌握及控制变更更加困难，最终将导致质量与规范问题。

掌握遗产系统消耗大量成本

"应用程序刚上线后不久便成为遗产系统。主要原因在于负责的开发人员的离去，导致该应用程序的详细知识蒸发。因此“遗产”这个术语也扩大到了当前最新开发语言。"^[2]

对于遗产系统的分析，修改及测试等软件开发生命周期中各作业所需时间将变得 longer，并且发生错误的频率高，容易失去系统实施及集成良机。

对现有应用系统的迅速正确的了解是成功进行系统升级或运维的重要因素之一。在没有确保对应用系统的可视性的情况下很难在有限的时间和成本内成功完成此类项目。

[2] "Strategies To Cut Application Costs And Increase Productivity Using Application Mining Tools" Phil Murphy, Forrester Research, 2009. 4.

应用系统文档 - 无用之物

遗产系统的本质问题是知识的遗失，开发结束后不久开发人员的离去或代码变更之后没有文档化，最终因为没有足够的对该应用程序经验丰富的专家或正确的文档导致费用增加，生产率低下。新投入的开发人员和设计人员必然无法快速掌握该应用系统，并且当修改源代码或数据库的某个部分时也无法掌握受影响的领域，另外测试人员对选定测试对象及测试范围变得困难。

应用程序的知识遗失之后，只能依赖同事或文档，但所有企业都因为不正确的应用程序文档所困惑着。因为文档与实际生产环境的应用程序内容不一致，大部分文档将变成无用之物。在对应用系统没有正确的知识的情况下，运维该应用系统可以说是一种噩梦。

应用程序自动分析工具提供的可视性

何为应用程序自动分析工具?

应用程序自动分析工具自动采集源代码和数据库对象，并通过解析(parsing)之后提取解析的各对象之间的所有关联关系和有意义的度量信息。提取出的信息保存到储存库中，最终构筑应用系统的知识库(knowledgebase)。

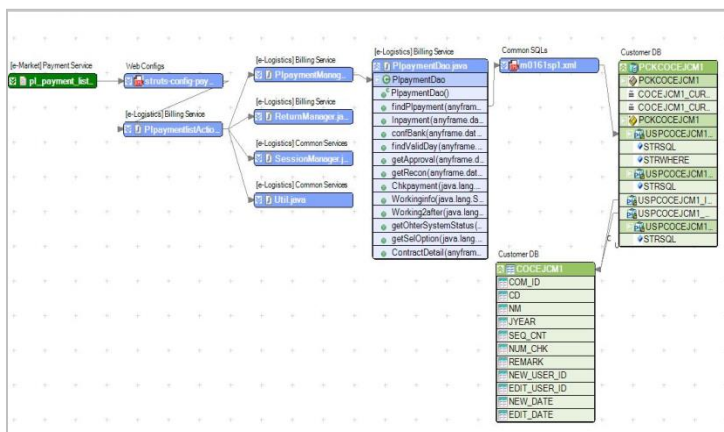


[图 1] 工具的基本概念

用户无需手动映射解析的各对象之间的关联关系或依赖关系。工具根据用户输入的基本设置全部自动执行。

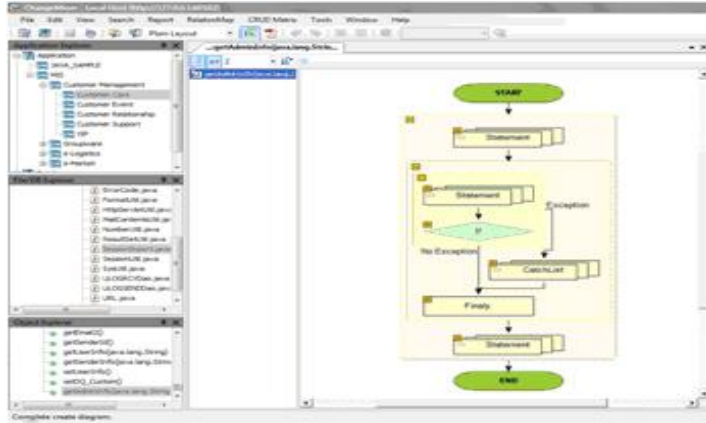
为提高用户的生产率，从应用程序知识库中可获得的信息如下：

- 调用关系(Calling chain) – 从客户端页面到数据库的全范围内的对象或函数相互调用信息。
- 视觉图(map) – 为提高对应用程序的了解，通过图展示所有对象之间的调用及被调用关系。



[图 2] 可视化关联图示例

- 整体依赖关系 – 对象或变量，数据库字段等依赖于其他领域的信息。
- 源代码程序流 – 对源代码文件中的所有执行路径通过流程图展示。



[图 3] 流程图示例

- CRUD 矩阵 – 通过矩阵展示程序和数据库之间数据的新增(Create), 查询(Read), 修改(Update), 删除(Delete) 相关信息。

Table	C	R	U	D
POKCOCECTM.LUSPCCOCECTM_INSERT				
POKCOCECTM.LUSPCCOCECTM_UPDATE				
POKCOCEXM1.LUSPCCOCEXM1_INSERT				
POKCOCEXM1.LUSPCCOCEXM1_UPDATE				
ORCL.REGRESS.ADBMS.D...				
POKCOCEFRM1.LUSPCCOCEFRM1_INSERT				
POKCOCEFRM1.LUSPCCOCEFRM1_UPDATE				
POKCOCEHSM1.LUSPCCOCEHSM1_INSERT				
POKCOCEHSM1.LUSPCCOCEHSM1_UPDATE				
POKCOCEJCM1.LUSPCCOCEJCM1_INSERT				
POKCOCEJCM1.LUSPCCOCEJCM1_UPDATE				

```

1 PACKAGE pckCOCEXM1 AS
2 TYPE COCEXM1_Cur01 IS REF CURSOR;
3 TYPE COCEXM1_Cur02 IS REF CURSOR;
4
5 -- 1. 환승관리 MASTER(공통-회계) 목록 가져오기 ...
6 PROCEDURE uspCOCEXM1_LIST(pCur OUT NOCOPY COCEXM1_Cur01,
7                          VSrchField IN VARCHAR2,
8                          vSearchDate IN DATE)
    
```

[图 4] 确认CRUD 矩阵及其相应代码行

- 多种度量 – 基于行业标准的复杂度和规模信息，无任何调用的代码，无任何引用的数据库字段，死代码等。

为什么需要?

导入应用程序自动分析工具的重要原因是提高整体应用程序相关的生产率，帮助IT部门用更少的资源处理更多的工作。

影响分析

应用程序自动分析工具向用户提供与变更的代码存在关联关系的因素，因此开发人员可以掌握修改应用程序及数据库时实际的范围。并且可以集中测试变更所涉及的影响范围领域。进行变更影响的开发人员因为充分考虑到了变更导致的影响，因此可提高正确性与信赖度。总而言之，通过影响分析IT部门将大幅降低频繁变更程序导致的系统故障。

缩短对应用程序的理解以及AS-IS分析时间

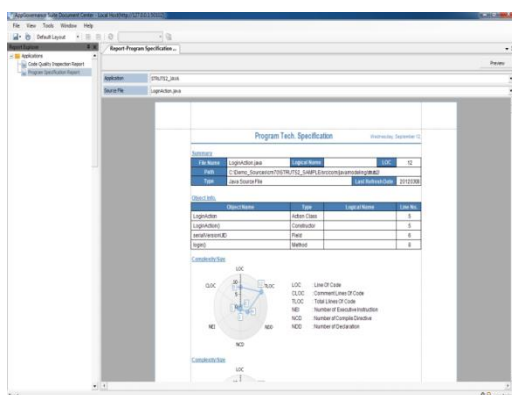
应用程序自动分析工具提供减少应用程序理解以及AS-IS分析所需的努力和费用的方法。工具的储存库如同相关人员随时可以查阅的词典，帮助快速理解应用程序的结构和关联性。特别是在系统迁移(migration)或更新(modernization)项目中，自动化替代人工作业从而可大幅缩短时间及减少费用。在没有确保应用程序可视性的情况下，对本人要作业的应用程序不熟悉的开发人员将依赖低效的应用程序分析及理解方法。例如：查阅老旧的文档，直接读代码或者找之前参与开发的人员进行咨询，这些都会增加项目费用。

更有效的协作及应用知识共享

没有应用知识储存库时，随着时间的流逝以及应用程序的变更，最初设计/实现阶段开始积累的知识将逐渐消失。应用程序自动分析工具将储存库中的信息以图形式可视化展示，帮助用户导航应用程序的各角落并可以向下跳转到相关数据和代码。因此当导入此工具时，项目组可以迅速简便地将相关知识传达给新投入的开发人员，并且企业通过可视化以及透明的应用知识共享提高开发人员，管理人员以及质量管理人员之间的协作能力。

文档化作业的改善

文档化一般指静态意义上的文档制作，往往此类文档在应用程序上线后会在短时间内被遗忘。代码变更之后不及时同步更新到静态的文档，因此类文档的正确性项目组处理日常业务时不会去使用它。应用程序自动分析工具持续地更新储存库，并提供生成自动化文档(动态文档化)生成工具。因此项目组可以始终使用最新的文档。



[图 5] 动态生成文档例



解决其他问题

应用程序自动分析工具在以下领域可以提供帮助:

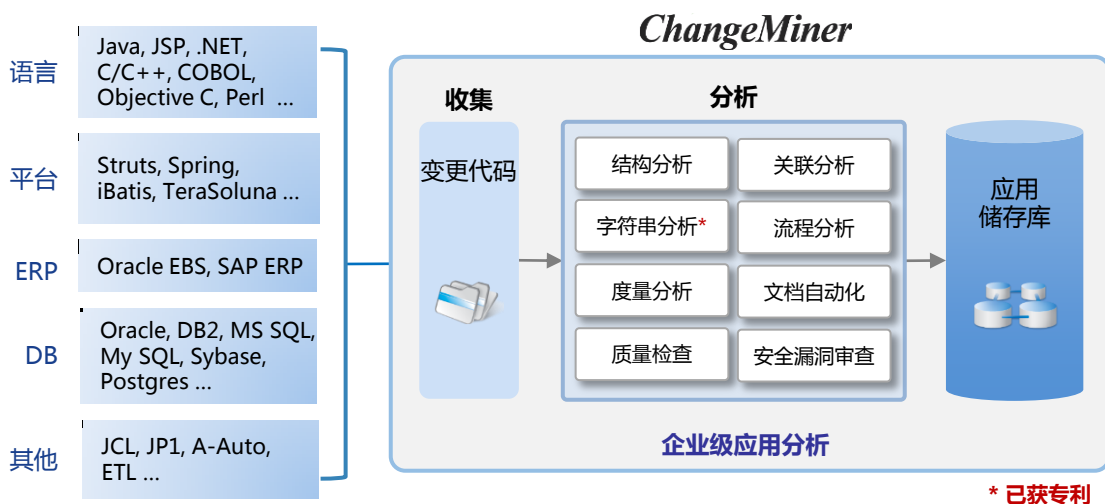
- 测定应用程序复杂度和规模 – 自动计算应用程序的复杂度和规模指标并提供数据。
- 开发人员遵守技术标准义务化 – 工具可检测未遵守技术标准的代码行，因此可验证开发人员编写的代码是否遵守企业规定的技术标准。开发人员可以事先检测代码质量及时应对将来有可能发生错误。
- 评价外包开发 – 通过此工具可以评价外包开发的应用程序的质量或遵守标准与否。
- 预测应用程序变更费用 – 工具提供需要变更代码所涉及的影响因素以及各因素之间的关联关系。因此可以通过变更影响列表比较客观地预测变更所需费用，另外可以减少变更导致发生系统故障的费用。
- 清除不使用的资源 – 在代码文件和数据库内部中找出完全没有调用关系的对象并清除，从而可以减少运维系统所需要付出的努力。
- 获取应用程序及数据库资产情况 – 管理人员可以随时查询到正确的应用程序及数据库的资产信息，例如各业务应用程序的代码文件数和代码行数，哪些业务包含复杂的模块，哪些业务程序和数据库经常变更等。

企业的应用管理者，开发者，设计及分析者，质量管理等所有相关人员都可以有效地使用应用程序自动分析工具。

ChangeMiner – 强大的应用程序自动分析工具

ChangeMiner是?

GTONE ChangeMiner是搭载完整功能的应用程序自动分析工具，其提供应用程序与数据库间完整的可视性。ChangeMiner以服务器，存储库，客户端3个要素组成。简单架构如下图所示。



[图 6] ChangeMiner 框架

第一个组成要素，ChangeMiner服务器基于J2EE技术，以提供对平台的独立性和稳定性。工具内置分析服务器以及视图服务器，分析服务器提供如下功能：

- 从远程计算机或本地计算机采集源代码以及数据库对象
- 解析及分析采集的源代码以及数据库对象，并提取结构信息以及关联信息
- 将提取的信息储存到中央存储库

视图服务器为J2EE Web应用程序，并提供如下功能：

- 处理来自ChangeMiner 客户端的请求 – 提供分析结果数据
- 处理来自ChangeMiner管理员控制台的请求 – 执行管理工作

第二个组成要素，ChangeMiner存储库为关系型数据库，用于储存事前分析好的所有数据，作为应用程序知识库，向开发人员和管理人员提供有用的信息。

第三个组成要素，ChangeMiner客户端基于.NET技术，使用户界面更加丰富。为方便用户使用，客户端支持零配置安装，并提供自动版本升级功能。用户可以随时随地通过互联网进行访问。

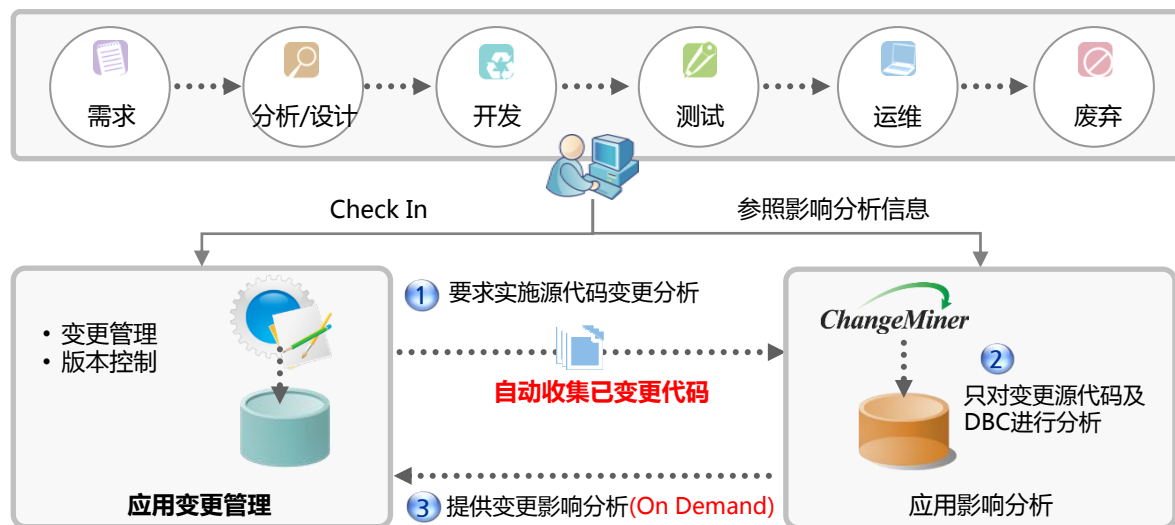
ChangeMiner用户可以通过客户端从应用程序存储库查询关联图、CRUD矩阵图、各种Diagram、功能点分数、报表等多样的信息。ChangeMiner还支持动态文档生成功能，使用户提高文档相关工作的生产性。

ChangeMiner 优势 – On-demand 分析

与其他应用程序自动分析工具不同，ChangeMiner支持只针对变更代码的On-demand分析。即使，识别变更代码，并只对变更代码进行解析，也可以准确分析发生变更的关联关系。

通常，其他工具为识别变更的关联关系，需要解析全部源代码，并反复进行扫描。因此，以Batch形式，定期分析全部的源代码。在这种情况下，会发生用户根据Batch周期，使用不是最新数据的情况。

ChangeMiner 支持 On-demand 形式分析以及Batch 形式分析。因此，确保其柔韧性的同时，提供最新的影响分析信息。ChangeMiner的这种先进的技术，也确保了ChangeMiner与应用程序生命周期管理工具的接近实时的集成。



[图 7] ALM与 On-demand 集成概念图

ChangeMiner 优势 – 动态字符串分析专利技术

为了分析程序源代码与数据库间的相互关系，ChangeMiner除了使用广泛所使用的语法解析技术以外，同时使用称为USA(Universal String Analyzer)的独特的专利技术。GTONE 为了克服不能通过基于语法解析而提取的源代码内固有的相互关联性数据，从而开发了USA。

USA可以在不执行程序的情况下，准确的提取动态字符串。假设，存在使用下图所示动态SQL语句的源代码：

```

72  ResultSet rs = null;
73
74  try {
75
76  con = ConnectionManager.getConnection();
77
78  StringBuffer sqlQuery = new StringBuffer();
79
80  sqlQuery.append("select count(*)    ") ;
81
82  if( kind == 1)
83      sqlQuery.append("    from action    ");
84  else
85      sqlQuery.append("    from behavior    ") ;
86
87  sqlQuery.append("where agent_id = '"+agent_id+"'    ")
88  sqlQuery.append("    and action_ste = '00'    ") ;
89
90  LogManager.addDEVLog(3, "sung", this.getClass().getName(),
91  "getAccidentHistory()", sqlQuery.toString()+"", "selectQry");
92
93  pstmt = con.prepareStatement(sqlQuery.toString());
94

```

[图 8] 简单动态 SQL 示例

USA 可以提取程序执行时，最终可能拼接出的动态字符串：

- Select count(*) from action where agent_id =? and action_ste=' 00' ;
- Select count(*) from behavior where agent_id =? and action_ste=' 00' ;

ChangeMiner 可以通过USA提取的上述两个动态语句，准确的分析出程序及数据库表之间的关系。

USA可以使用于源代码内所有字面值(Literal)。因此，ChangeMiner通过相应技术，提取关联性相关数据，如，服务名、COBOL程序ID、外部文件里的SQL语句ID值等。

USA 最显著的优点是降低分析结果的 False Positive。假如，没有动态字符串分析技术，如USA，用户需要执行额外的工作来整理大量False Positive数据。



支持大部分常用语言以及相应语言之间的关联关系

与单一语言分析工具或IDE插件不同，ChangeMiner 可以分析通过COBOL，Java，.NET等多用语言进行开发的业务应用程序之间的关系。

假设，业务应用程序的表示层通过JSP或Java开发，业务逻辑层为COBOL，数据库为MainFrame DB2。ChangeMiner 也可以分析出其全部关联关系，并将结果以直观的方式提供并将结果以直观的方式提供给用户。因此，也支持从企业整体观点，进行系统变更时，所需的影响分析及品质水平的分析。

ChangeMiner支持的语言以及数据库如下：

语言	Java, JSP, XML, Java Script, Objective C, C#.NET, VB.NET, ASP, PHP, Perl, Visual Basic, PowerBuilder, Delphi, ANSI C/C++, Pro*C, Amdocs C, PL/1, RPG, Visual Gen, COBOL: ZOS, MF, HITACHI, ILE
数据库	Oracle, DB2, MS-SQL Server, Informix, Sybase ASE, AS IQ, My SQL, Postgres
Framework	J2EE Hibernate, iBATIS, Struts, Spring, Terasoluna, Ladybird, Microsoft WCF, WPF ...
ERP	Oracle EBS (Forms & PL/SQL), SAP (ABAP)
Job Scheduler	JCL, JP1, A-AUTO
EAI	IBM MQ, Web Services, Tivoli, Shell for Informatica, DataStage, SAP BW, MSTR
其他	上述列表里没有记载的非常用语言，根据Universal Pattern Analyzer ^[3] 以及Handler ^[4] 也可以对其调用模式进行分析。

[3] Universal Pattern Analyzer 是基于用户可以输入的正则表达式以及扫描参考与否等技术，提取源代码内特定编程模式，分析关联关系的机制。

[4] Handler为一系列API的组合，支持用户通过解析出的对象信息进行更加详细的附加处理工作。

结论

应用程序自动分析工具提供具有深度的应用程序相关可视性，从而帮助专家解决应用相关多种问题。

应用程序自动分析工具通过采集、分析源代码以及数据库对象，提取全部关联信息，并提供有用的度量信息。

ChangeMiner是搭载完整功能的应用程序自动分析工具，提供对应用程序的完整的可视性。

ChangeMiner的独特性来自于以下特点:

- 以变更代码为对象执行的On-demand分析
- USA (Universal String Analyzer)专利技术 – 不执行程序，也可以准确的提取动态字符串
- 可以提取跨语言应用程序之间的所有关联关系

企业的应用开发及维护部门可以通过应用程序自动分析工具解决很多问题，从而减少应用相关费用、降低应用的复杂性。为了提高应用管理相关效率，仅仅对源代码进行版本管理或生命周期管理是远远不够的。现在，使用自动分析工具进行应用治理是非常必要的。



<http://www.gtonesoft.com>

<http://www.changeminer.com>

Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the GTO one License Agreement and may be used only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronics medium or machine readable form without prior consent, in writing, from GTO one, Corp.

Information in this document is subject to change without notice and does not represent a commitment on the part of GTO one. The software and documentation are provided "as is" without warranty of any kind including without limitation, any warranty of merchantability or fitness for a particular purpose. Future, GTO one does not warrant, guarantee, or make any representations regarding the use, or the results of the use, of the software or written material in terms of correctness, accuracy, reliability, or otherwise.