

发起人高层会议总结报告

RAGHAV S. NANDYAL
SITARA TECHNOLOGIES PVT. LTD.

Dear Raghav Nandyal:

一、总体发现

通过 CMMI Level 5 范围界定的评估，对软件开发和项目管理流程进行了全面而细致的审查，采用了 CMMI V3.0 (开发领域) 标准。在此期间，进行了文档审查和面谈，并且 HMLA 和 ATM 在评估中准确地识别了已知和潜在的问题。他们在评估期间提供了有针对性的改进建议，并进行了相关的指导。我们非常感谢 ATM 团队，他们使用幽默风趣的例子帮助我们更好地理解各种流程中的弱点，并进行了多次沟通和交流。同时，在基准评估中，评估团队除了软件流程外，还对我们的业务流程进行了全面审查，并在可选的参与者会议中使用 SPRUM-Systemic Process Review Using Measurements® (Raghavan S. Nandyal 的注册商标)，提供了对“如何进行有效的后续绩效改进计划”的更深入见解。在此次评估中，各位评估老师们从产品需求、设计、开发、测试、交付、决策分析、项目管理、项目资源的精细化管理、PPM 模型的引入、人才培养机制、编码规范等多个方面提出了宝贵的改进建议。

通过与 Raghav 的深入讨论，我们获得了宝贵的建议和指导，这为公司未来的发展奠定了基础。评估过程让我们进一步的深入审视了公司现行的 CMMI 高成熟度的实施现状。有了更为全面和深入的了解，发现了不少潜在的问题与改进空间。公司高层、EPG 过程改进小组、项目经理、质量保证、以及配置管理等核心成员，经过对这些问题的深入追溯与研讨，达成了共识，本次评估活动产生了非常实用的评估结果，将会帮助我公司在更多正在执行的项目中推广高成熟能力应用实践。通过本次评估，也锻炼了我们的团队，提升了员工对标准化、规范化流程的认识和执行能力。这些建议与公司实际情况高度吻合，符合度达到 94%。

我们将认真消化吸收评估结果，结合公司实际情况，制定具体的改进措施和计划，确保每一项建议都能得到有效落实。这次 CMMI5 评估审核不仅是对公司研发能力和管理水平的全面检验，更是一次难得的学习和提升的机会。我们将以此为契机，不断总结经验、优化流程、提升能力，推动公司在追求卓越、实现持续发展的道路上迈出更加坚实的步伐。

二、经验教训

接下来，我们从工程过程、项目管理、支持过程、过程管理四个方面为切入点分别进行详细总结。

工程过程

在工程过程管理控制中，重点在设计编码方面，各位专家老师和评估老师给到了非常详细且落地的执行方案，例如：

部分在重用代码，特别是来自内部或组织内部以外的源代码（例如，BEsSoftware Corporation 公共引用），这些代码已有 10 年以上的历史（例如，2016 年），必须“进行审查，

以确保很久以前的代码仍然值得重用，或者是否应该重新编写以确保遗留代码风格与最新的编码风格相匹配。

针对这个问题，我们需要对代码进行全面测试和检查，确保带阿米的功能仍然符合当前需求。对代码进行性能分析和调优，了解其运行效率和资源使用率，对代码进行扫描，确保代码没有安全漏洞，检查代码风格是否与当前团队的编码规范一致，如果不一致，必要时，可以进行优化和代码重构，同时对代码追加注释，确保功能简单明了，我们也会对代码定期进行兼容性测试，确保旧代码可以适用新的操作平台、浏览器、数据库等，另外，我们将同步检查代码文档是最新的，确保能准确反映出代码的功能和代码的实现逻辑。

在大多数代码中，变量名和变量类型相似，导致在必须考虑更改时难以阅读、维护和升级。针对这个问题，我们将完善升级我们的编码规范，定义有意义的变量名，确保变量名能准确的表达用途或所存储的数据，而且必须用全名，不允许使用缩写，缩写的变量会增加理解困难度；变量的命名必须遵循行业通用的驼峰式命名方法，而局部变量则必须使用下分割线，而对于类成员变量，则要求变量名前加上特定的前缀，用来区分是全局变量还是局部变量。除此之外，我们还将制定检查单，并定期对代码进行检查，以便及时指出命名不当或者难以理解的地方，确保代码具有可读性和可维护性。对经常犯错的团队成员，我们将给与一定的善意惩罚，而对于代码一直较好的则给与奖金表扬。

而对于测试策略中未指定回归测试的条件和先决条件，尤其是在对 IT 基础架构进行配置更改时。验证和确认在不同测试活动期间所使用的测试用例不明确的问题，我们将在团队中引入测试集的概念，在每轮测试前，我们将根据本轮测试范围，评估出关联的测试用例，并从测试用例池中，导入这部分测试用例作为本轮回归的用例范围，再此基础上，我们确定测试用例执行的先后顺序，以满足测试用例的前置条件。

项目管理

在数据质量检查中，数据检查只标注了结果，并未对数据质量要求的原因，和检查时观测的信息进行登记，也没有使用适当的测量数据库来确保数据完整性，由于此类数据用于规划和估算项目参数的统计和定量管理，因此必须尽量减少易受人为因素影响的数据输入错误。

这个问题我们也注意到了，这类问题分别影响项目的计划阶段和过程监控阶段，甚至还影响到结项分析阶段，针对这个问题，我们将建立公司数据质量标准，并定期进行检查，设定专门的人员兼职进行检查和质量评审，并计划在项目管理平台中，专门建立数据统计表进行存储数据，为保持数据不会被人为改动，我们将设定人员读写权限，基线数据已经通知发布，将重置数据状态，不再支持修改。

另外一个问题是关于风险：与过程管理相关的风险，必须与项目或产品特定风险同等重要性对待。目前，风险管理的重点集中在：需求风险、计划风险、组织与管理风险、人员风险、环境与客户风险。然而，过程风险的管理似乎太过粗略，难以满足过程管理（PCM）和过程资产开发（PAD）的需求。

风险管理的粗略性可能会导致项目中的潜在问题被忽视或处理不当，针对风险管理和风险措施的问题，我们讨论后，一致认为，需要有明确的风险管理流程，并在识别风险后，建立风险跟踪矩阵，甚至要预测风险，通过这类手段来识别风险系数和优先级，同时我们也将增加风险课程培训，强调团队成员对风险管理的认识和重视，鼓励团队成员积极识别风险和报告潜在

风险，对风险制定的措施，要经过团队的演练模拟，要确保风险措施具有可操作性和目标明确性，并约定风险实施人员完成时间，另外，对重大风险隐患，鼓励项目管理者上报高层，确保高层领导对风险管理给予足够的重视和支持，避免风险被隐藏。

支持过程

首先，严格执行编码标准。用户定义类型(例如，`InputLifecycle` 类型的 `lifecycle` 和 `inputLifecycle`)的“类似 `read`”变量声明的几个实例，从方法调用(例如，`map.get(input.getId ())`)获取输出并被分配，这使得代码极其模糊，难以阅读维护和增强。

这类问题首先要求质量保证人员应深入学习和理解所使用的编程语言及其编码规范。掌握变量命名、类型声明、方法调用等方面的最佳实践，确保进行代码审查时，保持高度的敏锐度，注意识别潜在的代码可读性和可维护性问题，对用户定义类型、变量命名、方法调用等关键部分进行细致审查。而质量保证人员还应该制定详细的审计流程和检查清单，确保覆盖代码的所有关键部分。在审计流程中增加对代码可读性和可维护性的专门检查环节。同时，质量保证人员需要与开发团队建立紧密的合作关系，共同制定符合公司真实场景的代码规范和最佳实践。并且，质量保证人员，需要被邀请参与开发过程中的代码评审会议，及时提出改进建议。最后，在代码审计过程中，质量保证人员要注意识别可能导致代码可读性和可维护性问题的潜在风险。对识别出的风险进行评估和分类，制定相应的应对措施。定期对风险进行监控和评估，确保其得到有效控制。如发现风险有增大趋势，应及时调整应对措施并加强监控力度。

另外，在基础设施方面，组织项目指标数据库具有许多定义措施，用于分析。表格本身必须受到密码的保护，以确保数据损坏最小化。考虑在数据库管理应用程序中实现数据捕获，存储和检索以限制访问控制。

我们也正在考虑建立公司数据质量标准，并定期进行检查，设定专门的人员兼职进行检查和质量评审，并计划在项目管理平台中，专门建立数据统计表进行存储数据，为保持数据不会被人为改动，我们将设定人员读写权限，基线数据已经通知发布，将重置数据状态，不再支持修改。

过程管理

尽管对需求阶段的评审工作进行了原因分析，但也要充分考虑以下 2 个影响因素，第一需要从投入和产出比上研究，否有必要这样早的阶段做阶段分析。第二，对于确定的关键原因，制定的改进措施颗粒度还应该进一步细致。

针对这个问题，我们讲收集与重大问题相关的投入和产出数据，包括人力、物力、财力等资源的投入，以及产品、服务、利润等产出的数据。对数据进行分析，计算投入产出比，了解资源利用效率和经济活动的效益情况，然后估当前投入产出比的合理性。识别投入与产出之间的不匹配或低效环节，定期监控和评估投入产出比的变化情况，对实施方案进行技术、经济、资源等方面的可行性评估，确保实施方案在技术上是可行的，经济上是有益的，资源上是充足的，并且明确各执行主体的职责和分工，确保任务明确、责任到人，接着制定详细的实施计

划，包括时间节点、任务安排、资源分配，确保实施计划具有可操作性和可追踪性，对实施方案的执行情况进行定期监督和评估，及时发现和解决问题，确保实施方案能够按计划顺利推进，并保持各执行主体之间的沟通和协作，确保信息共享和协同工作，提高实施方案的执行效率。

工作辅助工具（模板、检查单和项目过程）中存在重复的流程信息，且其格式与现行或可用的实践做法不同。例如：过程定义中的纠正措施请求（CAR）触发条件，必须与用于落实 CAR 实践的清单无缝衔接。需要仔细研究这些额外的内容，并将其减至最少。

针对这个问题我们将仔细审查现有的模板、检查单和项目过程文档，标记出所有重复出现的流程信息。向使用这些工具的员工或团队收集反馈，了解他们是否也注意到了重复信息，并询问他们对解决这个问题的看法。了解重复信息是如何产生的，是由于不同的工具之间缺乏协调，还是由于某个工具在更新时未能及时删除旧的信息？分析重复信息对工作效率、资源利用和准确性等方面的影响。制定标准化的流程描述，确保所有工具中的流程信息都是一致和准确的，建立一个有效的更新机制，确保当流程发生变化时，所有相关的工具都能得到及时更新，以避免旧信息的残留，向我们的员工提供培训，确保他们了解新的工具和使用方法，并知道如何避免在未来产生重复信息。建立一个持续的反馈循环，鼓励员工在使用工具时提出任何发现的问题或改进建议。根据收集到的反馈和业务流程的变化，定期更新工具，以确保它们始终与当前的工作需求保持一致。

我们根据过程管理方面提出的一些改进建议，EPG 小组也进行了详细的讨论和分析，制定了相应的措施。比如过程管理方面提出了一些针对组织的改进计划目标、内容及其他，应该随着改进的推进来优化和调整，后续，我们将致力于加强数据分析和业务逻辑之间的融合，通过更深入的研究和探索，完善影响盈利能力的依赖关系定义。同时，我们也会积极与团队成员分享这次的经验教训，共同提升我们在数据分析方面的能力和水平，为公司的盈利增长提供更有力的支持。

三、现实意义

实施 CMMI5 高成熟能力模型带来的现实意义非常显著。

1、CMMI5 的实施推动了企业对软件开发过程的持续改进和量化管理。通过对软件开发过程进行精细化的控制和度量，企业能够及时发现并解决开发过程中的瓶颈和问题，提升开发效率和质量。同时，量化管理使得企业能够对开发进度、成本和质量进行更加准确的预测和控制，为企业的决策提供有力支持。

2、CMMI 高级成熟度鼓励跨团队的协作和沟通，促进知识共享，增强组织内部的协作文化。通过实现高级成熟度，组织能够在竞争中脱颖而出，展示其卓越的过程能力和产品质量，增强市场地位。

3、CMMI5 的实施推动了自动化程序在软件开发中的应用。基于规则和量化分析的自动化程序可以大大提高软件开发的效率，减少人为失误，使得软件开发过程更加稳定、可控。这不仅有助于提升企业的研发能力，还能够为企业带来更高的经济效益。

4、CMMI 5 高级成熟度的组织通常能够提供更高质量的产品和服务，这直接影响客户的满意度和忠诚度，进而提高市场竞争力。高级成熟度要求组织在管理过程中依赖定量数据。这使得决策更加基于事实而非直觉，提高了决策的科学性和准确性。

5、实施 CMMI5 高成熟能力模型有助于企业在软件研发和管理领域达到业界领先水平。这代表着企业已经建立了一套成熟、高效、规范的软件开发和工程管理流程，能够在复杂多变的市场环境中迅速应对各种挑战，提升企业的竞争力和市场占有率。

6、CMMI5 适用于大规模、高质量、复杂度高的软件开发项目。通过实施 CMMI5，企业能够确保软件项目的顺利进行和高质量交付，满足客户的期望和需求。这不仅有助于提升企业的市场声誉和品牌形象，还能够为企业带来更多的商业机会和合作伙伴。

7、CMMI5 强调过程自我修正能力的提高，有助于企业降低软件开发风险。通过不断优化和完善软件开发过程，企业能够减少错误和缺陷的产生，提高产品的稳定性和可靠性。这不仅有助于提升客户满意度，还能够为企业节省大量的维护成本和修复成本。

实施 CMMI5 高成熟能力模型对于我们公司来说具有深远的现实意义。它不仅有助于提升我司的软件研发和管理水平，还能够为我司带来更高的竞争力和商业利益。因此，我们会积极拥抱 CMMI5，不断推动自身的改进和发展。

四、改进措施

在本次评估后，EPG 团队将与项目人员精诚合作，一起对评估中提出的建议进行深入讨论与识别。通过会议交流和数据分析，我们将总结出差距分析材料，这份材料将成为我们现阶段组织过程改进的核心依据。

实施 CMMI5 的过程就是一个不断追求完善、不断优化的旅程，我们始终秉持着不断进取、永不停歇的精神。通过培训提升团队技能、制定标准化流程、强化监控机制、加强内部沟通以及持续改进等多种方式，我们致力于提升组织的能力与成熟度，确保我们能够适应不断变化的软件发展环境和市场需求。我们对人员的技能要求也在日益提高，因为只有不断提升自身能力，才能更好地满足客户的需求，提高开发效率，从而在激烈的市场竞争中立于不败之地。

EPG 团队将与项目人员合作，对评估中提出的建议进行深入讨论和识别，通过会议交流和数据分析，总结差距分析材料，作为改进的核心依据，制定详细的改进计划，包括步骤、时间安排、参与人员和实施目标，公司高层承诺提供必要的资源支持，确保改进工作的顺利进行，EPG 团队将实施改进措施，并保持全面监控，记录效果，进行量化分析，选取项目试点验证改进措施，通过假设检验后推广至整个组织，通过培训、标准化流程、监控机制和沟通，不断提升组织能力和成熟度，感谢评估师和 ATM 老师的指导，承诺根据发现进行改进，提高软件开发效率和质量，此次的评估让我们对 CMMI5 有了更为深刻的理解，同时也让我们认识到自身的不足之处。

基于这份材料，我们将精心制定《过程改进建议表》和《过程改进计划表》，其中将明确改进活动的每一步骤、时间安排、参与人员以及实施目标等关键信息。为确保改进工作的顺利进行，公司高层将与 EPG 成员共同讨论改进过程中可能出现的注意事项、潜在风险及制约点，并承诺提供充分的人、财、物等方面资源支撑。我们将通过制定规范、调配资源以及迭代生产工具等手段，及时解决这些问题，为改进工作创造有利条件。

在改进过程中，素有人员会保持高度的配合度。EPG 团队会根据发现的问题和分析报告，进行深入的根本分析，寻找有效的改进措施。后期，我们将选取合适的项目试点，对改进措施进行实际尝试验证。在试点数据通过假设检验后，我们将进一步推广部署这些改进措施，以实现更大的组织效益提升。

最后，我们衷心感谢评估师和各位 ATM 老师的辛勤付出和专业指导。感谢评估组提出的发现，我们很认同这些发现，感谢评估师和评估团队给出的宝贵意见。您们宝贵的意见为公司团队指明了前进的方向，为公司未来的发展奠定了坚实的基础。这些发现对于我们提高软件开发效率和质量，有很大帮助；后续我们会针对各项发现，认真落实改进，通过改进助力研发水平提升。持续提高软件开发效率和质量，提高客户满意，实现可持续发展。

我在此授权并同意您本人和 SITARA Technologies 在 SITARA 的出版渠道上分享我们的评估成果，在 SITARA Technologies 认为合适的情况下宣传我们的评估成果。


北京宝兰德软件股份有限公司
发起人：李洪巍
2025 年 1 月 18 日

EXECUTIVE SESSION BRIEFING: SPONSOR FEEDBACK

Overall findings

Through the benchmark appraisal defined by CMMI Level 5, the software development and project management processes of our R&D department were thoroughly and meticulously reviewed, based on the CMMI V3.0 (Development domain). During this period, document reviews and interviews were conducted, and HMLA and ATMs accurately identified known and potential issues during the appraisal. They provided targeted improvement suggestions and offered relevant guidance. We are very grateful to the ATM team for the weaknesses in various processes and engaged in multiple communications and discussions with us.

Additionally, during the benchmark appraisal, the appraisal team not only reviewed our software engineering processes but also conducted a comprehensive review of our business processes. In the optional executive management meeting, they used SPRUM-Systemic Process Review Using Measurements® (Raghavan S. Nandyal's registered trademark) to provide deeper insights into "how to carry out effective follow-up performance improvement plans." During this appraisal, the appraiser offered valuable improvement suggestions on various aspects, including product requirements, design, development, testing, delivery, decision analysis, project management, refined management of project resources, the introduction of the PPM model, talent training mechanisms, and coding standards.

Through in-depth discussions with Raghav, we received valuable advice and guidance, which laid the foundation for the company's future development. The appraisal process allowed us to further scrutinize the current implementation status of CMMI HM within the company. With a more comprehensive and in-depth understanding, we identified numerous potential issues and areas for improvement. After in-depth tracing and discussions of these issues by senior management, the EPG process improvement group, project managers, quality assurance, and configuration management core members, a consensus was reached. The results of this appraisal are highly practical and will help our company promote high-maturity capability application practices in more ongoing projects.

This appraisal also strengthened our team and enhanced employees' understanding and execution capabilities regarding standardized and regulated processes. These suggestions highly align with the company's actual situation, with a compliance rate of 94%.

We will carefully digest and absorb the appraisal results, and based on the company's actual conditions, develop specific improvement measures and plans to ensure the effective implementation of each suggestion. This CMMI Level 5 appraisal is not only a comprehensive examination of the company's R&D capabilities and management levels but also a rare learning and enhancement opportunity. We will use this as an opportunity to continually summarize experiences, optimize processes, enhance capabilities, and drive the company to take more solid steps in its pursuit of excellence and continuous development.

Engineering management

In the management and control of engineering processes, the focus is on design and coding aspects. Experts and appraisals have provided very detailed and practical execution plans. For example, regarding code reuse, particularly code from internal or external sources (e.g., public references from BES Software Corporation), some of this code is over 10 years old (e.g., from 2016). It must be reviewed to ensure that old code is still worth reusing, or whether it should be rewritten to ensure that the legacy code style aligns with the latest coding standards.

To address this issue, we need to conduct comprehensive testing and inspection of the code to ensure that the associated functionalities still meet current requirements. We will perform performance analysis and optimization to understand its running efficiency and resource usage, conduct code scanning to ensure there are no security vulnerabilities, and check whether the coding style aligns with the current team's coding standards. If discrepancies are found, we will optimize and refactor the code as needed, adding comments to make the functionality clear. We will also perform regular compatibility testing to ensure the old code is suitable for new operating platforms, browsers, databases, etc. Additionally, we will check if the code documentation is up to date to accurately reflect the functionality and implementation logic of the code.

In many parts of the code, variable names and types are similar, which makes it difficult to read, maintain, and upgrade when changes are required. To address this, we will improve and upgrade our coding standards, defining meaningful variable names to ensure they accurately reflect their purpose or the data they store. Abbreviations will be avoided, as they increase the difficulty of understanding. Variables will follow the industry-standard camelCase naming convention, with local variables using underscores, and class member variables requiring a specific prefix to distinguish them as global or local variables. Furthermore, we will develop a checklist and regularly review the code to identify inappropriate or unclear naming, ensuring that the code remains readable and maintainable. For team members who frequently make mistakes, we will apply mild penalties, while those with consistently good code will be rewarded with bonuses or recognition.

Regarding the testing strategy, there were unclear conditions and prerequisites for regression testing, especially when making configuration changes to IT infrastructure. We will introduce the concept of test sets within the team. Before each round of testing, we will appraise the relevant test cases based on the testing scope, and import them from the test case pool as the scope for regression testing. On this basis, we will determine the execution order of the test cases to meet their prerequisites.

Project management

In data quality checks, the process currently only marks the results without documenting the reasons for data quality requirements or recording the observed information during the checks. Additionally, appropriate measurement databases are not being used to ensure data integrity. Since this data is used for planning and estimating project parameters for statistical and quantitative management, it is essential to minimize data entry errors that are susceptible to human factors.

We have also noticed this issue, which affects the planning phase, process monitoring phase, and even the closing analysis phase of the project. To address this, we will establish company-wide data quality standards and conduct regular inspections. We will assign dedicated personnel to carry out checks and quality reviews, and we plan to create specific data tables within the project management platform to store data. To prevent data from being altered by unauthorized personnel, we will set read/write permissions, and once baseline data is published, it will be reset to a state that no longer supports modifications.

Another issue concerns risk management: risks related to process management must be treated with equal importance to project or product-specific risks. Currently, the focus of risk management is on: requirement risks, schedule risks, organizational and management risks, personnel risks, environmental and client risks. However, process risks are managed too roughly, which makes it difficult to meet the needs of process management (PCM) and process asset development (PAD).

The roughness in risk management may lead to potential issues in the project being overlooked or mishandled. After discussing this, we all agree that a clear risk management process is needed. Once risks are identified, we will establish a risk tracking matrix and even forecast risks. Through these methods, we can identify risk factors and priorities. We will also increase risk management training to emphasize the importance of risk awareness among team members, encouraging them to actively identify and report potential risks. Measures for managing risks will be tested and simulated by the team to ensure that they are operational and have clear objectives. We will also specify the responsible persons for risk implementation and their completion deadlines. For significant risks, we encourage project managers to report them to senior leadership to ensure that the leadership gives sufficient attention and support to risk management, thereby preventing risks from being overlooked.

Support

First, strict adherence to coding standards is essential. There are several instances of "read-alike" variable declarations in user-defined types (e.g., lifecycle and inputLifecycle of the InputLifecycle type), where outputs are obtained from method calls (e.g., map.get(input.getId())) and assigned. This practice makes the code extremely obscure, difficult to read, maintain, and enhance.

This issue requires quality assurance personnel to thoroughly learn and understand the programming languages used, along with their coding standards. They should master best practices in variable naming, type declaration, method calls, and more. During code reviews, they must remain highly vigilant and be able to spot potential issues with code readability and maintainability. Special attention should be given to user-defined types, variable names, method calls, and other critical sections. Quality assurance personnel should also create detailed auditing processes and checklists to ensure that all key parts of the code are covered. Specific checks for readability and maintainability should be added to the auditing process. Additionally, quality assurance personnel need to establish close cooperation with the development team to jointly develop coding standards and best practices that reflect the real-world scenarios of the company. Furthermore, quality assurance personnel should be invited to participate in code review meetings during the development process, where they can offer timely improvement suggestions. Finally, during code audits, quality assurance personnel must identify potential risks that could impact the readability and maintainability of the code. Once identified, these risks should be assessed and categorized, with corresponding mitigation strategies developed. Regular monitoring and evaluation of these risks should be carried out to ensure they are effectively controlled. If the risk is increasing, countermeasures should be adjusted promptly, and monitoring efforts should be strengthened.

Additionally, in terms of infrastructure, the organization's project metrics database has many defined measures for analysis. The database itself must be password-protected to minimize the risk of data corruption. It is also worth considering implementing data capture, storage, and retrieval within the database management application to restrict access controls.

We are also considering establishing company-wide data quality standards and conducting regular checks. We will designate dedicated personnel to conduct audits and quality reviews, and plan to create specific data tables within the project management platform to store the data. To prevent data from being tampered with, we will set read/write permissions. Once baseline data is published, its state will be reset, and modifications will no longer be supported.

Process Management

Despite conducting a cause analysis during the requirements phase, it's important to consider two additional factors. First, we need to study the cost-benefit ratio to determine whether it's necessary to perform such phase analyses so early. Second, the granularity of the improvement measures for identified key causes should be further refined.

To address this, we will collect input and output data related to major issues, including resource inputs like manpower, materials, and financial resources, as well as outputs like products, services, and profits. We'll analyze the data and calculate the input-output ratio to assess resource utilization efficiency and the economic benefits of activities. We will then evaluate the rationality of the current input-output ratio, identify mismatches or inefficiencies, and periodically monitor changes in the ratio. Feasibility assessments on the implementation plans will be conducted in terms of technical, economic, and resource availability, ensuring that the plans are technically feasible, economically beneficial, and resource-adequate. Clear responsibilities and task assignments will be defined to ensure clarity, accountability, and traceability in the implementation plan. Regular monitoring and evaluation will help ensure smooth progress and address issues as they arise, fostering communication and collaboration among stakeholders to improve execution efficiency.

Regarding work auxiliary tools (templates, checklists, and project processes), we have observed redundant process information, and the format does not align with existing or available practices. For example, the trigger conditions for Corrective Action Requests (CAR) in the process definitions must seamlessly integrate with the checklist used for implementing CAR practices. We will carefully review existing templates, checklists, and project documentation to identify all redundant process information. Feedback will be collected from employees or teams using these tools to understand whether they have noticed these redundancies and to gather their input on how to resolve the issue. We will analyze how redundancy arises—whether due to a lack of coordination between tools or because outdated information was not removed during tool updates—and assess its impact on work efficiency, resource utilization, and accuracy. A standardized process description will be developed to ensure consistency and accuracy across all tools. We will establish an effective update mechanism to ensure that when process changes occur, all relevant tools are updated promptly to avoid residual outdated information. Employees will be trained to understand new tools and methods and to prevent future redundancy. A continuous feedback loop will be established to encourage employees to report any issues or suggestions for improvement, ensuring tools remain aligned with current work needs.

Based on the improvements suggested in process management, the EPG group has had detailed discussions and analyses and developed corresponding measures. For instance, improvement plans targeting the organization should be optimized and adjusted as progress is made. Moving forward, the company will focus on strengthening the integration of data analysis with business logic, exploring deeper connections, and refining the definitions of dependencies affecting profitability. We will also actively share lessons learned with team members, improving our collective data analysis capabilities to better support the company's profitability growth.

Relevance

The practical significance of implementing the CMMI HM (High Maturity) model is substantial. Here are several key points:

Continuous Improvement and Quantitative Management: The implementation fosters continuous improvement in software development processes through detailed control and measurement. By refining the development process, the company can promptly identify and resolve bottlenecks, leading to enhanced efficiency and quality. Quantitative management enables more accurate forecasting and control of progress, cost, and quality, providing strong support for decision-making. **Fostering Collaboration and Knowledge Sharing:** High maturity encourages collaboration across teams, promotes knowledge sharing, and strengthens the organization's internal culture of cooperation. Achieving high maturity allows the organization to differentiate itself in the market by demonstrating superior process capabilities and product quality, thereby enhancing its market position. **Advancement in Automation:** The implementation accelerates the use of automated processes in software development. Automation, driven by rules and quantitative analysis, greatly enhances development efficiency, reduces human errors, and stabilizes the development process. This not only boosts the company's R&D capabilities but also contributes to greater economic benefits. **Improved Product Quality and Customer Satisfaction:** High-maturity organizations are typically able to deliver products and services of higher quality, which directly influences customer satisfaction and loyalty, thereby improving market competitiveness. High maturity requires organizations to rely on quantitative data in management, making decision-making more data-driven, scientific, and accurate. **Industry Leadership:** Implementing the CMMI HM model helps the company reach an industry-leading level in software development and management. It indicates that the company has established mature, efficient, and standardized development and engineering management processes, allowing it to rapidly adapt to challenges in a dynamic market, increasing competitiveness and market share. **Suited for Complex, High-Quality Projects:** The HM model is ideal for large-scale, high-quality, and complex software development projects. It ensures that these projects progress smoothly and are delivered with high quality, meeting customer expectations and requirements. This not only enhances the company's market reputation and brand image but also creates more business opportunities and potential partnerships. **Enhancing Risk Management:** CMMI HM emphasizes improving process self-correction capabilities, helping reduce software development risks. By continually optimizing and refining the development process, the company can minimize errors and defects, thereby improving product stability and reliability. This leads to increased customer satisfaction and substantial savings in maintenance and repair costs.

For our company, the implementation of the HM model has far-reaching significance. It will not only elevate our software development and management capabilities but also bring higher competitiveness and business benefits. Therefore, we are committed to embracing CMMI Level 5 and continuously driving our own improvement and growth.

Improvement measures

After this appraisal, the EPG team will work closely with project personnel to engage in in-depth discussions and identify the suggestions provided in the appraisal. Through meetings and data analysis, we will

summarize the gap analysis materials, which will serve as the core basis for our current organizational process improvement.

Implementing CMMI Level 5 is a continuous journey of striving for excellence and constant optimization. We remain committed to progress and improvement. Through training to enhance team skills, establishing standardized processes, strengthening monitoring mechanisms, improving internal communication, and maintaining a focus on continuous improvement, we aim to elevate the organization's capabilities and maturity. This will ensure we remain adaptable to the ever-changing software development environment and market demands. Our expectations for team skills are also continuously increasing, as only by constantly improving our capabilities can we better meet customer needs, improve development efficiency, and maintain a competitive edge in the market.

The EPG team will collaborate with project personnel to deeply discuss and identify the suggestions raised in the appraisal. By exchanging ideas in meetings and analyzing data, we will summarize the gap analysis materials to form the core basis for improvement. A detailed improvement plan will be created, which will include steps, timelines, participants, and implementation objectives. Senior management is committed to providing the necessary resources to ensure the smooth implementation of the improvement activities. The EPG team will carry out the improvement measures, maintain comprehensive monitoring, document the effects, and perform quantitative analysis. Pilot projects will be selected to validate the improvements, and after hypothesis testing, the changes will be rolled out across the entire organization. Through training, standardized processes, monitoring mechanisms, and communication, we will continually improve the organization's capabilities and maturity. We would like to thank HMLA and ATM for their guidance, and we commit to making improvements based on the findings to enhance software development efficiency and quality. This appraisal has given us a deeper understanding of CMMI Level 5 and has highlighted areas where we need to improve.

Based on this material, we will carefully prepare the "Process Improvement Suggestions Table" and the "Process Improvement Plan Table," which will clearly outline the steps, timelines, participants, and implementation goals of the improvement activities. To ensure the smooth progress of these efforts, senior management will work with EPG members to discuss potential issues, risks, and constraints that may arise during the improvement process. They are committed to providing sufficient resources, including personnel, budget, and equipment, to support the improvements. We will address these challenges by establishing standards, reallocating resources, and iterating production tools to create favorable conditions for the improvement work.

Throughout the improvement process, team members will maintain a high level of cooperation. The EPG team will conduct in-depth root cause analysis based on identified issues and reports to find effective improvement measures. Later, we will select suitable pilot projects to practically validate the improvements. After testing the pilot data through hypothesis testing, we will further deploy these improvements to achieve greater organizational benefits.

Finally, we sincerely thank HMLA and all the ATMs for their hard work and professional guidance. We appreciate the discoveries made by the appraisal team, which we fully agree with. We are grateful for the

valuable suggestions provided by HMLA and ATMs. Your insights have pointed our company's team in the right direction and laid a strong foundation for the company's future development. These findings are extremely helpful in improving software development efficiency and quality. Moving forward, we will carefully implement improvements based on these discoveries to enhance our R&D capabilities. We are committed to continuously improving software development efficiency and quality, increasing customer satisfaction, and achieving sustainable development.

As the sponsor of this appraisal who has received the executive session briefing, I hereby authorize and give consent to you and SITARA Technologies to share our appraisal accomplishments on SITARA's publishing channels giving publicity to our appraisal accomplishment as SITARA Technologies deems it fit.

BEIJING BAOLANDE SOFTWARE CORPORATION

Sponsor: Hongwei.Li

January 18, 2025

