

企业软件配置管理实施策略

毫无疑问，软件配置管理是软件开发的基石。一个缺少软件配置管理的企业，就等于“作坊式”的开发模式。虽然大多数企业在这一点的认识上已经达成共识，然而，在实际的实施时仍存在不少误区。正确地在企业实施软件配置管理，有赖于全面正确地认识软件配置管理，以及避免一些常见的误区，并选择正确的工具。

由于配置管理涉及的领域既有广度，又有深度，因此，本演讲不打算面面俱到，而只是抽取其中的一部分，希望能起到抛砖引玉的效果。

正确认识软件配置管理

要避免在软件配置管理实施时的误区，首先需要我们对软件配置管理的含义有一个正确全面的认识，在这一方面，基本上有两个权威的解释，一个是 SEI，一个 CMM 方面的。SEI 的定义要点包括：

- ✓ 配置管理（CM）是软件过程的一个关键元素。它是一个规程，通过控制产品的进化过程，如持续的、变化的变更，为软件系统产品提供了稳定性。
- ✓ 作为一个管理规程，CM 通过标识产品的组成部分和变更；通过控制变更的开始、评估、授权和控制；通过记录和报告产品及其变更的历史和状态，最终控制了产品的整个进化过程。
- ✓ 作为一项开发支持功能，CM 用来维护产品的实际组成部分；记录产品的组成部分乃至整个产品的变化历史；为产品的变更提供了一个稳定的工作平台；支持从产品的组件“组装”成整个产品；并自动协调并发的修改。

CMM/CMMI 的定义要点包括：

- ✓ 软件配置管理的目标是在软件的整个生命周期期间建立和维护软件产品的一致性。软件配置管理包括：标识软件项目的配置项；控制配置项的修改；记录和报告配置项的状态和修改活动。
- ✓ 配置管理的核心是意图消除由于不同版本工件的存在而可能带来的混乱和错误。项目中工件的变化不可避免，原因可能包括纠正错误，功能增强以及产品的进化。配置管理就是要使得这些不可避免的变更处于控制之下。

大家可以看到，无论是 SEI 的定义，还是 CMM/CMMI 的定义，二者之间并没有本质上的区别。归纳起来，从总体上来说，软件配置管理主要包括五个主要方面，即配置项的**标识**、对配置项修改的**控制**、向团队成员**报告**软件配置管理的状态、**审计**软件配置管理活动、实现自动化的**构建与发布**，其中第五点更是与时下流行的敏捷趋势伴随而流行。

全面认识软件配置管理的含义，可以帮助我们认知在企业内实施配置管理时的先后步骤，以循序渐进的方式来实施。另外，也可以帮助我们全面管理企业实施时所涉及的活动。

在软件配置管理的五个主要方面中，很明显，标识是基础，即首要的第一步是要确定哪些对象需要纳入到配置管理的控制之下，候选的对象包括：源代码，设计文件，用户手册，数据库脚本，构建脚本，网站图形元素等等；接下来需要确定如何控制对这些配置项的修改，包括环境的搭建，用户授权，开发流程等等；随后，要及时向团队成员报告软件配置管理的状态，履行告知的义务，以及进行审计，确认相关的软件配置管理活动确实按照预定的计划高质量地完成了。

认识到这五个方面的软件配置管理活动的相互关系非常重要，它们之间互为前提。同时，认识到这一点也告诉我们在进行软件配置管理建设时不要好高骛远，而是要分阶段一点一点的稳步前进。

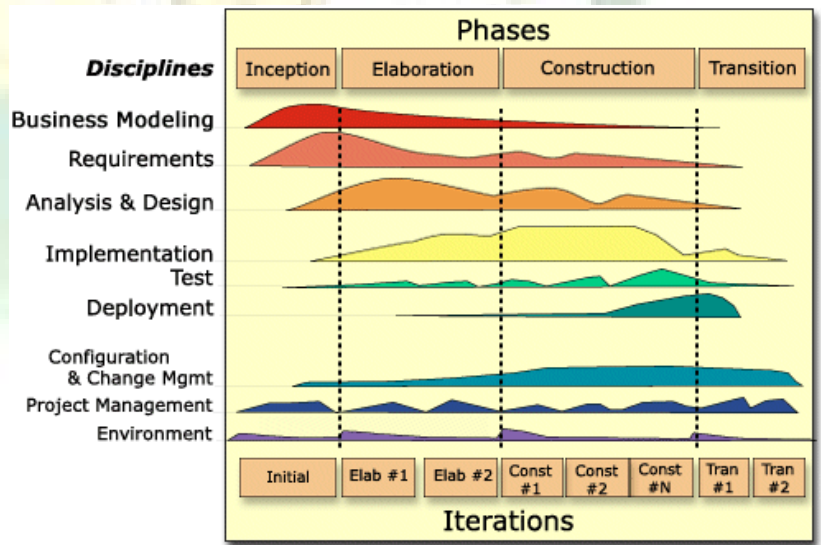
这五个方面，软件配置管理工具都要进行强有力的支持，使得日常事务减至最少，这是一个成熟的软件配置管理工具应具备的基本特征，将在后面讲述。

如果您到此还对上面的定义感到太学术的话，我们不妨换一个角度来理解软件配置管理。实际的软件开发活动中，常常会有以下的“课题”，请您想想该如何解决呢？

- ❖ 如何跟踪供应商（Open Source）的代码？
- ❖ 如何管理并行开发？
- ❖ 如何进行分布式开发？
- ❖ 如何实现自动化的发布？
- ❖ 如何实现变更管理的流程自动化？
- ❖

如果一下子回答不上来的话，也没关系。这些问题都是软件配置管理研究的范畴，掌握软件管理方面的理论，选择合适的工具，以及采用合适的实施策略，将会使这些问题一一得以解决。

由此可见，软件配置管理在软件开发（研发）活动中具有基础性的地位，无论是 Borland 的 ALM 理念，还是 RUP 的开发流程，以及敏捷开发，都对这一观念进行了充分的认可。



CMM/CMMI 对软件配置管理的活动进行了进一步的细分，如下图所示：

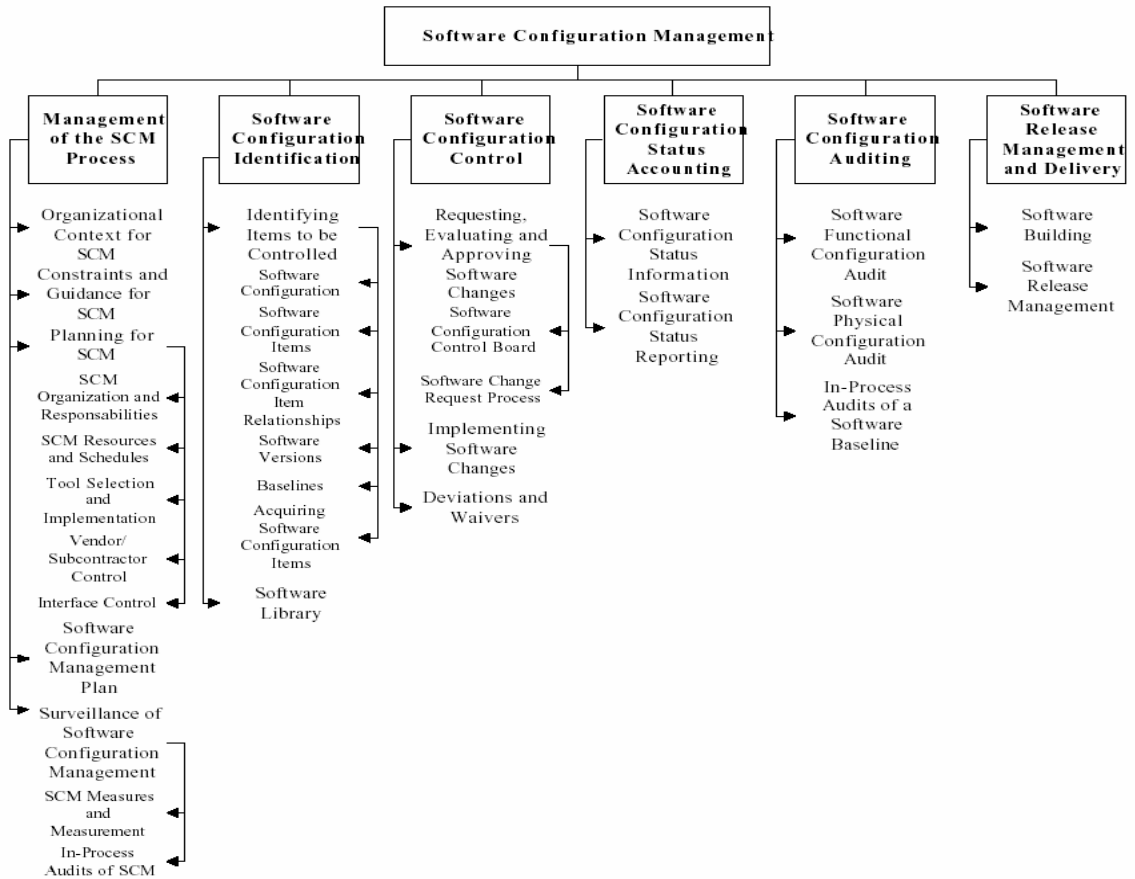


Figure 2 Breakdown of SCM Topics

从图中可见，CMM/CMMI 将软件配置管理的活动分成了六个方面，每个方面又再进行了细分。除了上述的五个方面外，还加上了实施软件配置管理所需要的组织架构上的支持活动。

全面认识软件配置管理的活动，可以防止在实施的范围上有所偏差或遗漏。

组织架构与实施级别

前面提到，为了有效实施配置管理，还需要组织架构上的支持活动，这就需要具备一定的组织结构。视企业的规模以及实际应用开发时的情形，软件配置管理常常在三个级别上实施：项目级，应用级和组织级，不同级别的软件配置管理需要履行不同的职责内容，因而也具有相应的组织结构，如下表所示：

软件配置管理 实施级别	具体内容	履行角色	涉众/受益者
项目级	<ul style="list-style-type: none"> ● 将配置管理任务纳入到项目总体计划之中 ● 对代码打基线，作为新开发的稳定基础 ● 建立适当的分支策略与结构 ● 编译代码 ● 创建带交付物的发布包 ● 参加 CCB 会议 	项目中的兼职或专职配置管理员	项目经理及项目组成员
应用级（产品级）	<ul style="list-style-type: none"> ● 评估应用（产品）级别的配置管理需求 ● 选择最适合该应用的配置管理技术 ● 定义应用级别的配置管理计划（独立地），如果在 	专职配置管理员、配置管理经理	产品经理

	组织级别不存在的话 <ul style="list-style-type: none"> ● 为应用建立配置管理环境 ● 为应用建立配置管理流程（检出、检入、构建、发布、变更控制、问题管理等等） ● 执行配置管理培训 		
组织级	<ul style="list-style-type: none"> ● 确定配置管理知识域 ● 评估组织级别的配置管理风险 ● 评估配置管理的支持和赞助者 ● 定义 SCM 预算 ● 建立配置管理的针对性架构 ● 定义组织级别的配置管理计划 ● 定义通用配置管理术语 	配置管理组, 包括组长, 发布经理等等	组织管理层

认识到这一点非常重要，无论是组织还是个人，在实施软件配置管理时首先要清楚自己实施软件配置管理时的范围是在项目级，还是应用级，或是组织级。不同级别的活动重心不一样，认识到这一点可以使得将有限的资源投入到关键的地方，从而确保成功。

实施误区及对策

尽管国内在软件配置管理的应用上已经有着多年的历史，然而，失败的情况仍然比比皆是，原因很多，这里仅列出其中的一些情形，希望能起到警示的作用。我们在此不谈论由于缺少软件配置管理而带来的现象，大多数企业已经不再在是否实施软件配置管理方面犹豫不决。尽管如此，有了主观上的愿望还远远不够，企业在实施软件配置管理时还需要注意相应的策略以及防范可能会走入的误区。

1. 没有专职的 SCM 人员；没有兼职的 SCM 人员；没有 SCM 人员

这是一个很常见的现象，由于对配置管理的内容认识不清，组织常常不重视配置管理，项目组除了项目经理，常常是清一色的开发人员。缺乏配置管理人员或“凑合”的配置管理人员常常会造成项目的诸多问题，如版本丢失，问题的重复出现，并行开发力度不够等等。

2. SCM 人员地位不高；团队整体 SCM 意识不够

地位不高的 SCM 人员缺乏推行 SCM 制度的力度，团队的 SCM 意识不够更是造成实施的难度加倍。

3. 对 SCM 缺乏正确全面的认识

1). 范围

对 SCM 实施的范围缺乏全面的认识常常使得实施时一叶障目，将焦点集中在有限的几个点上，例如版本控制和 Bug 管理，而其他方面却成为了盲点，例如软件的构建与发布。

2). 偏见

在主观上对配置管理存在一些偏见，认为它过分强调过程与文档，过于繁琐，并且也解决不了项目中的最终问题。

其实，配置管理是一项管理活动，是管理活动就得付出一定的管理成本，只要将配置管理的活动和前面所描述的相应级别相适应，其投资回报也将最大化，为此所付出的成本是值得的。

另一方面，项目的成功有赖于诸多方面的努力，配置管理只是其中一方面。

4. 实施大多以失败告终

1. 缺少中高层支持
2. 缺少必要的培训与支持，团队对实施的目标认识不清
3. 不够敏捷
4. 没有把握好流程、人员、工具三者之间的关系

5. 没有做好配置审计、报告环节

成熟软件配置管理工具的特征

相对于其他 CASE 工具，配置管理工具应该是最必不可少的，它可以帮助你管理软件开发时繁琐的工作。从早期的基于文件的版本控制工具，如 CVS，到今天现代的软件配置管理工具，如 ClearCase、StarTeam 等，软件配置管理工具已经有了长足的发展，并且依然在快速的发展着。软件配置管理工具发展过程中的关键特征如下表所示：

第 1 代：基于文件，以版本控制、支持 Check out/Check in 模型和简单分支为主要特征

第 2 代：基于项目库，支持并行开发团队协作以及过程管理

第 3 代：全面结合 CM 管理等各个软件开发环节的软件配置管理整体解决方案

企业要实施软件配置管理常常面临的第一步就是要选择合适的工具，在此将列出一个成熟的软件配置管理工具应该具备的特征：

- ❖ 配置项（对象）管理
 - 版本控制
 - 配置管理
 - 并行开发支持
 - 基线支持
- ❖ 构建与发布管理
 - 能利用流行的构建工具:ANT/MAKE
 - 支持多平台构建
 - 支持并行构建
 - 能自动处理构建依赖关系
 - 能收集和维护重新产生之前构建所需要的信息
- ❖ 工作空间管理
 - 能自动跟踪工作空间中所有类型的变更
 - 能应用不同配置填充工作空间
 - 工作空间既允许隔离又允许更新
- ❖ 流程管理
 - 不同类型的对象都应具备流程定制能力
 - 流程的范围可定制
 - 支持测试与发布流程
- ❖ 分布式开发的支持
 - 负载均衡
- ❖ 与其他工具的集成能力
 - 变更请求工具
 - 开发工具
 - 其他 CASE 工具
 - 命令行，SDK
- ❖ 易用性、易管理性
 - 报告能力

- 架构的弹性

Borland StarTeam2006 演示

作为新一代的软件配置管理工具，尽管历史不长，StarTeam 却有着众多的好评。随着 2006 版本的发布，StarTeam 在易用性，功能的全面性上仍在进行着长足的发展。

StarTeam 简介

StarTeam 作为一个配置管理工具，能够对产品和项目的整个开发生命周期进行有力的管理，主要包括两大方面

一、提供快速而有效率的：

1、版本控制

StarTeam 能对各种配置项类型进行自动化的版本管理，并且在版本分支上也能实现自动化。

2、过程管理

StarTeam 对项目开发过程的支持堪称完美，通过在项目级强制过程规则的使用，可以实现基于需求的开发、基于变更请求的开发、以及基于任务的开发等开发模式，并且，通过定制 workflow，可以以技术手段强制过程的采纳。

3、构建（Build）管理

通过与 ANT, CruiseControl 等开源构建工具，或诸如 BuidForge 等商业工具的集成，可以实现完美的构建管理能力。

4、并行开发

StarTeam 的并行开发通过视图机制来实现。

二、提供了一序列的支持功能

1、项目（Project）和视图（View）管理

视图是 StarTeam 的核心内容，StarTeam 提供了多种类型的视图，从而对并行开发、权限配置、发布等提供了灵活的支持。

2、变更跟踪

StarTeam 集成了变更管理功能，这一点是它区别于其他很多商业配置管理工具的地方。

ChangeRequest 组件可以实现与 TestDirector 组件的集成，由于 TestDirector 主要的用户为测试部门，通过集成可以实现测试部门与开发部门间跨部门的协作。

3、主题讨论

4、任务管理

5、仓库（Repository）定制，允许你添加新的属性到文件、变更请求、主题和任务上

6、对 workflow 进行定制以适应组织的需要，强制环境标准

基于完整而具有弹性的功能，StarTeam 简化了软件开发过程的管理，从而帮助完成配置管理任务，它所带来的收益主要体现在如下方面：

一、改善了开发流程

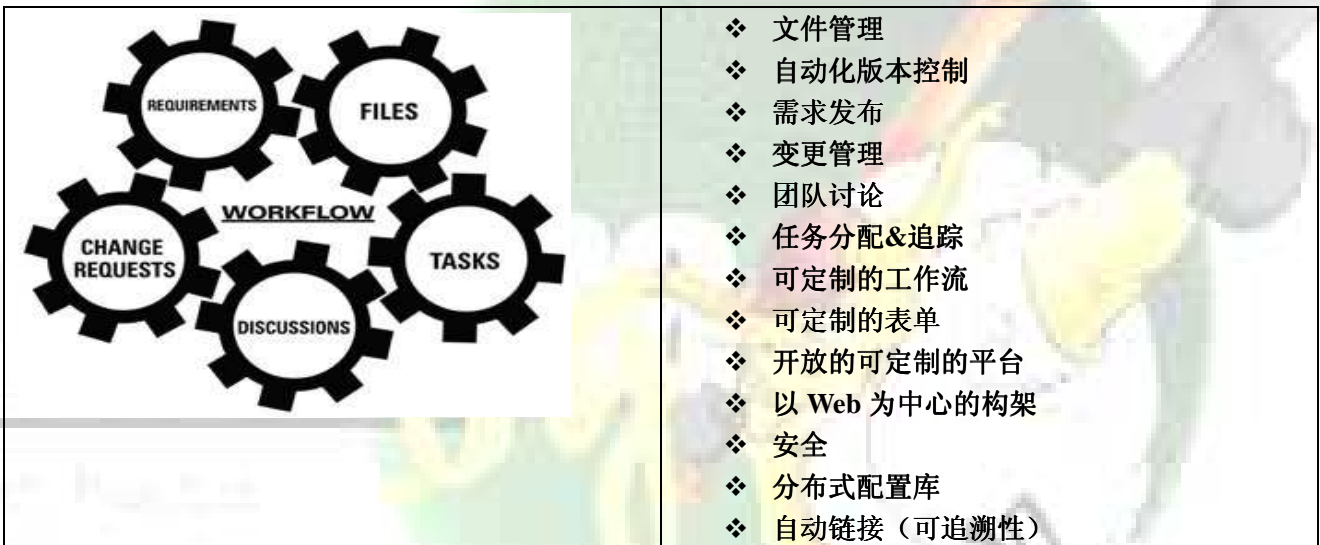
1、增强了团队成员间的沟通质量

2、提高了估计过程的准确度

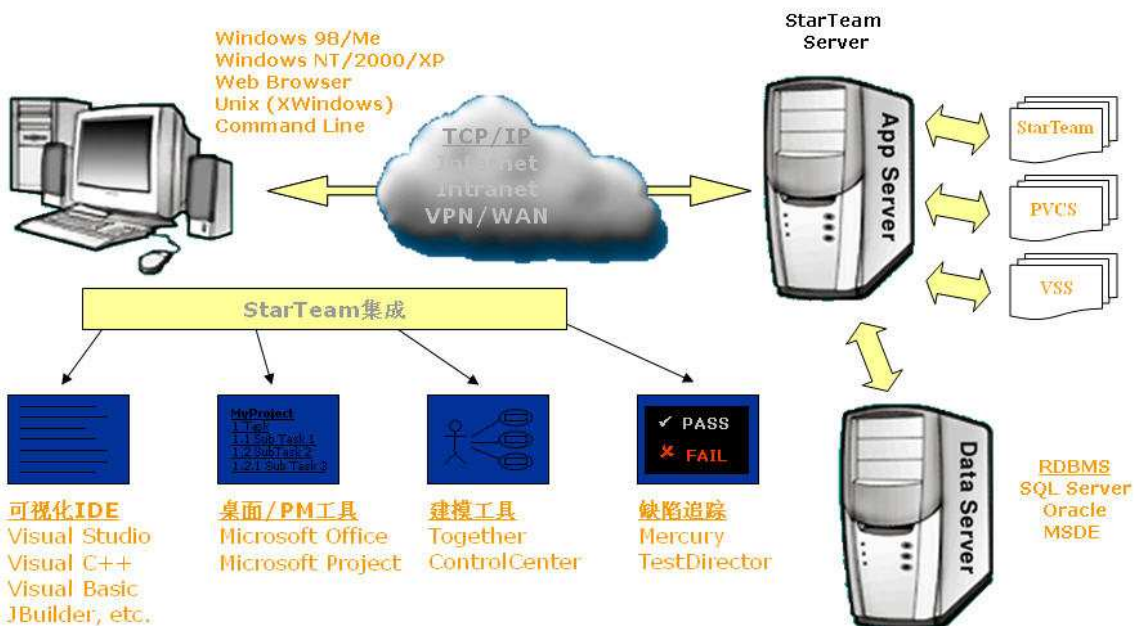
3、改善了在开发、测试、开发循环中项目管理信息流转的质量

- 二、加速了开发过程
 - 1、减少任务间的流逝时间
 - 2、减少了不必要的会议与备忘录
 - 3、自动化软件打包处理
- 三、强制标准的执行
 - 1、清晰、自然的版本控制过程
 - 2、轻松定制 workflow 和窗体

StarTeam 核心功能



StarTeam 架构 - 以 Web 为中心的架构



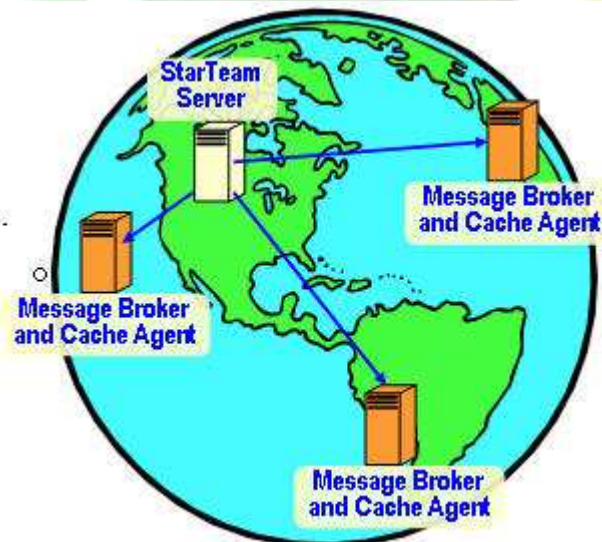
上图展示了 StarTeam 的基本架构,但并不代表 StarTeam 的全貌,例如 MPX 在上图中并没有体现出来,它将在随后讲述。

图中的 StarTeam Server 控制了所有对数据的访问,以防止数据被病毒感染。它使用当前流行的数据库,如 Oracle, SQL Server, 因而没有为了掌握一个专有的数据库产品而产生的陡峭的学习曲线和昂贵的管理成本。

StarTeam 有多个不同类型的客户端接口,用来支持不同的平台和不同的用户,你可以远程使用任意客户端接口,因为它们与服务器之间的通信都是通过 TCP/IP 协议进行的。很多配置管理产品不具有这一能力,它们可能只允许你远程使用 Web 浏览器的方式进行访问,这意味着你将不能在你自己的 IDE 环境中使用该产品。

StarTeam 可以与 VSS 和 PVCS 库进行集成,这对于遗留系统来说是非常有用处的,这些遗留系统仍然可以与当前的 CM 系统一起协同运行,这样,你将仍然可以使用已有的构建过程,而这些文件来自何处(到底是 StarTeam, 还是 VSS 或 PVCS)对用户来说是透明的。这样,系统就可以被逐步地按照需要被移植过来,从而使得由于使用新技术所带来的影响最小化了。

StarTeam 可以与许多其他产品进行集成,一个新的集成可以很容易地使用 SDK 创建出来。客户端与服务器的通讯采用开放标准的 TCP/IP 协议,并且传输过程可以压缩和加密。



StarTeamMPX 组件将 StarTeam 带入了支持分布式开发的领域。StarTeamMPX 是 Enterprise Advantage 版的组成部分。它为 StarTeam 服务器提供了一个发布/订阅方式的消息架构。通过对客户机/服务器框架进行扩展,StarTeamMPX 加速了信息在客户端和服务器的传输速率。通过使用高级客户端缓存,事件被实时地发布到客户端。通过自动将最新的信息传递给用户,使用户对数据的访问更快速,同时也由于本地缓存的使用,消除了大量的与服务器的通信,因为很多数据已经存在于本地缓存了。

StarTeamMPX 2006 由下列部件组成:

- Event Transmitter
- File Transmitter
- Message Broker
- Multicast Service
- Root and Remote Cache Agents

StarTeam2006 新特性

- ❖ 公共特性

- 原子 Check In
- 支持大文件
- 新的文件夹/文件比较/合并工具 替代了 Visual Diff/Visual Merge
- ❖ 服务器
 - 添加了 Checkout Trace Utility
 - 添加了针对 Native-II 格式仓库的 Vault Verify Utility
 - 定制自动通知 Email 的内容
 - 允许用户修改过期的口令
- ❖ 跨平台客户端
 - 嵌入式的文件比较/合并工具
 - 新的视图比较/合并工具
 - 与 Windows 客户端类似的特性
- ❖ 跨平台客户端（续）
 - 新的菜单命令
 - ❖ 最近访问的项目
 - ❖ StarTeam 日志
 - 个性化选项对话框中增加了一些选项
 - Layout Designer
 - 定制组件标签页的出现顺序
 - 定制 Detail 窗格的显示
 - History,Label 和 Reference 窗格中增加了新的上下文菜单命令
 - 右上窗格中添加了一个 Child Folder 标签页
 - Bulk Check-Out 实用程序提供了更多新特性
- ❖ 其他
 - 新创建项目根视图的 Branch On Change 属性缺省为选中状态
 - 在回滚视图中也能 Detach/attach 标签

