3DEXPERIENCE®

软件协同数据管理

DASSAULT SYSTEMES | The 3DEXPERIENCE® Company

ENOVIA

基于ENOVIA的软件协同设计与数据管理

- **Connected Software Engineer：**
- 在多学科产品开发背景下实施软件的协作、连接环境

包含的Apps	前置角色
 Connected Software	 Collaborative Business Innovator  Collaborative Industry Innovator

©Dassault Systèmes | Confidential Information | 1/13/2025 | ref.:3DS_Document_20013

概述

- Connected Software Engineer使工程师能够在3DEXPERIENCE平台上创建、管理和治理软件源内容。
- 它提供一个DesignSync和GIT连接器，将源代码和构建元数据从DesignSync和GIT SCM*工具（分别）联合到平台上。
- 系统和软件工程师可以将软件开发内容连接到平台上，以便在一个整体的硬件/软件定义中发布和管理。
- 连接的软件工程师提供在多学科产品结构中管理软件源代码和构建工件的能力，以实现可追溯性和治理。
- 该角色提供了在多学科产品背景下管理软件的能力。当与各自的连接者一起使用时，软件工程师可以引用外部SCM工具（如GIT或DesignSync）中管理的软件内容。
- 软件工程师能够管理和支配软件构建工件，将工件与软件源代码和构建配方联系起来，以生成构建。
- 软件工程师可以在多学科产品的背景下管理软件工程项目的可变性和有效性。

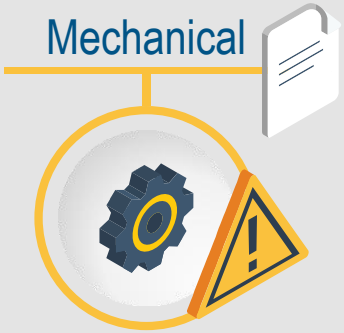
效益

- 通过改善多学科设计团队之间的协作来加速创新。
- 获得透明度，使代码审查、知识转移和根本原因分析得到改善。
- 减少返工，因为开发团队中的每个人都可以快速看到针对软件项目发布提交的最新软件内容。
- 通过将软件开发与产品开发过程联系起来，提高开发效率，使软件与产品战略更好地结合起来。
- 通过整合业务驱动的软件开发与需求、问题和变更管理，更有效地解决问题、履行变更单和完成任务。
- 通过一个单一的平台来获取和完善硬件和软件组件的需求，提高产品质量。

亮点

- 通过将软件作为完整的多学科产品配置的一部分，改善多学科合作。
- 通过纳入企业治理和流程，包括变更、问题和需求管理，对软件和硬件流程进行统一治理，从而执行最佳实践。
- 在3DEXPERIENCE上进行高效和协作的基于网络的代码审查。
- 获得对提交的最新软件内容的可见性，以改进代码审查、知识转移和根本原因分析。
- 确保与上游和下游流程的整合和连接，包括变更、问题和需求管理。

软件工程师的挑战



↓ Productivity and Innovation

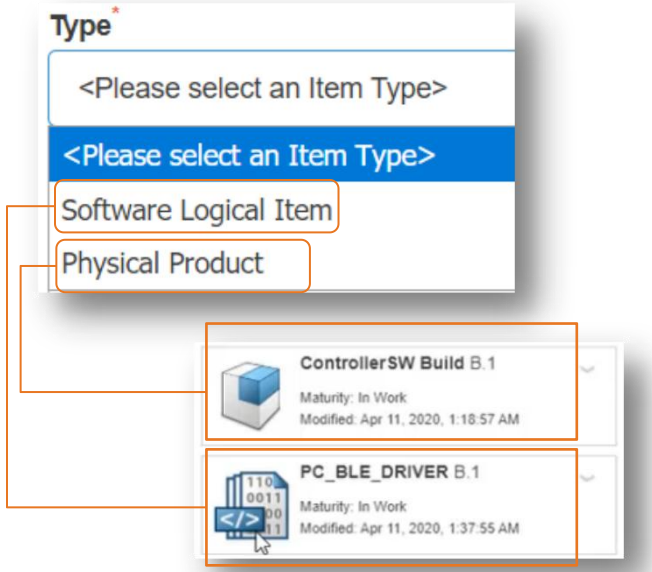
✕ Change control

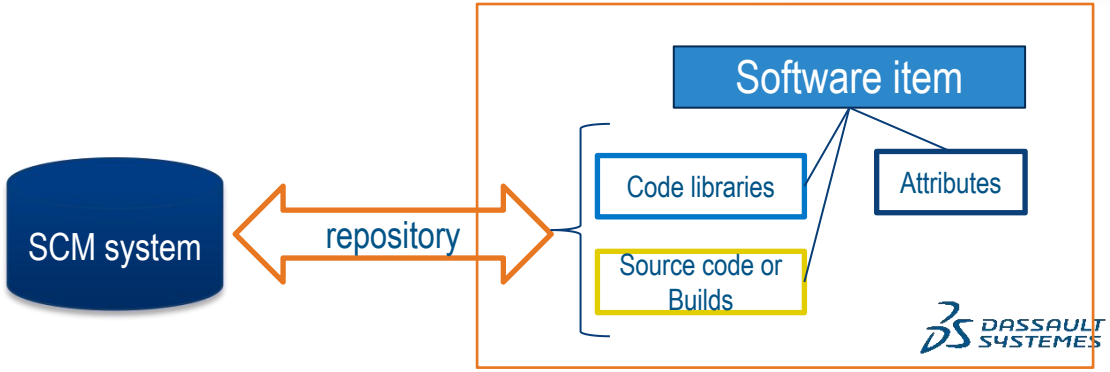
✕ Issue management

✕ Traceable requirements

什么是 "软件item"?

- 软件item是指包含修订控制下的软件内容的工程项目。
- 软件项目的类型可以是 :
  - 软件逻辑Item
    - 一个想法的集合，如源代码
  - 物理产品
    - 是软件的物理形式，如软件构建（编译的可执行文件、库和其他类型的物理元素）。
- 软件item通过人工或变更管理过程来管理代码/构建的生命周期.
- 软件item可以在平台上创建，使用 :
  - 网络上的 "连接的软件 "小工具
  - 本地CATIA 3DEXPERIENCE应用
- 软件item是由以下因素驱动的 :
  - 需求
  - 问题
  - 变更流程
  - 任务





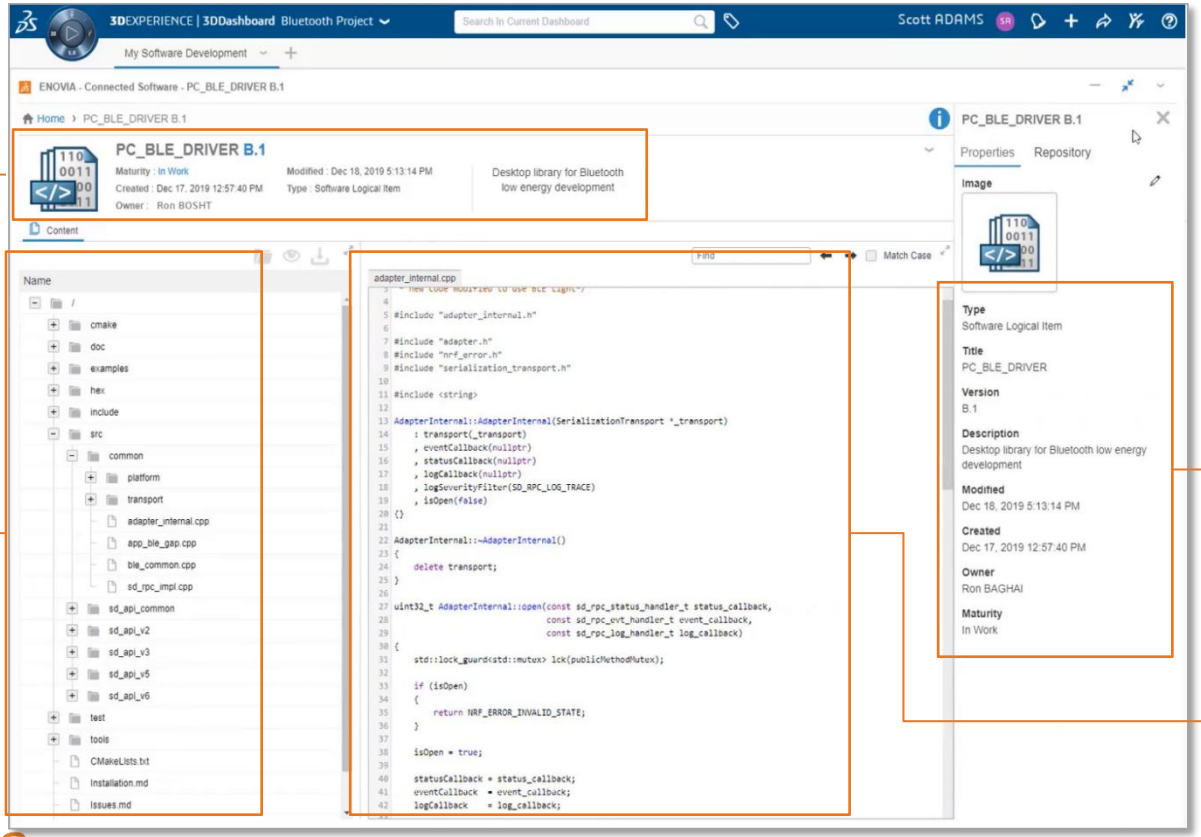
连接软件应用程序

Attributes

Library structure (from external SCM)

Properties

Source code (from external SCM)

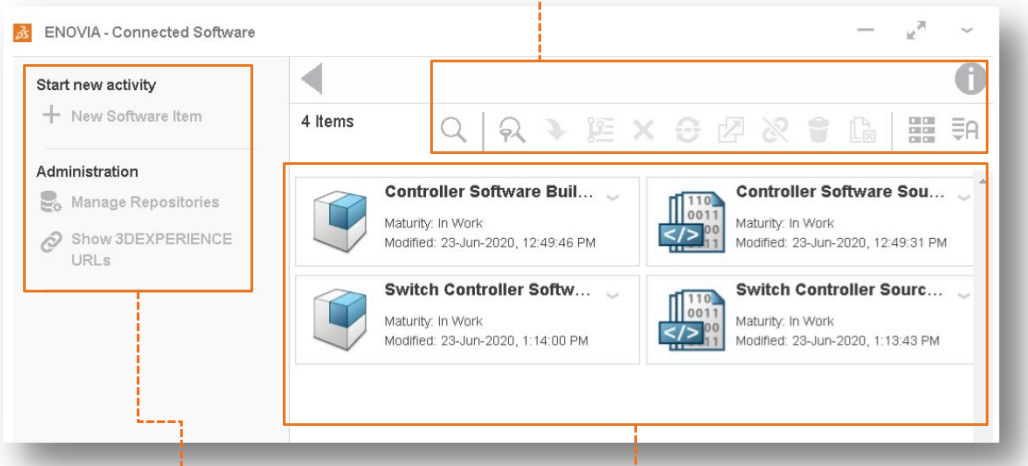


ENOVIA

DASSAULT SYSTEMES

主页 - 连接软件应用

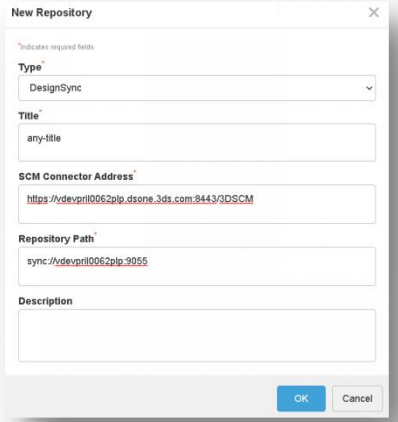
- 连接的软件小组件的主页显示
  - 左侧面板包含快速创建/管理软件相关数据的命令
    - 新建软件Item
      - 创建新的逻辑item或物理产品
    - 管理存储库
      - 创建或管理软件库。
    - 显示3DExperience URLs
      - 揭示3DExperience服务器的URLs
      - 设置外部版本控制系统与3DEXPERIENCE平台之间的连接时需要这些URL
  - 小部件的右侧包含
    - 所有软件项目的item
    - 在列表的顶部，有几个额外的命令对浏览所有item很有用



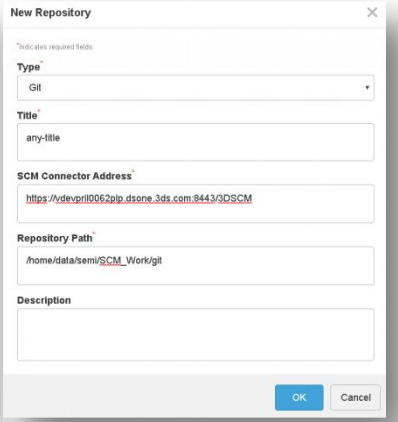
什么是软件item"库"?

- 在该平台上创建的存储库将软件item与软件开发的外部SCM系统连接起来。
- 这些资源库存储的信息允许从外部SCM系统定位和获取软件数据到平台。
  - 这使得在任何时候都可以使用基于网络的、对语言敏感的查看器来查看最新的软件代码。
- 平台中的一个存储库可以连接到以下任何一个SCM系统：
  - GIT
  - DesignSync

Repository definition for DesignSync



Repository definition for GIT

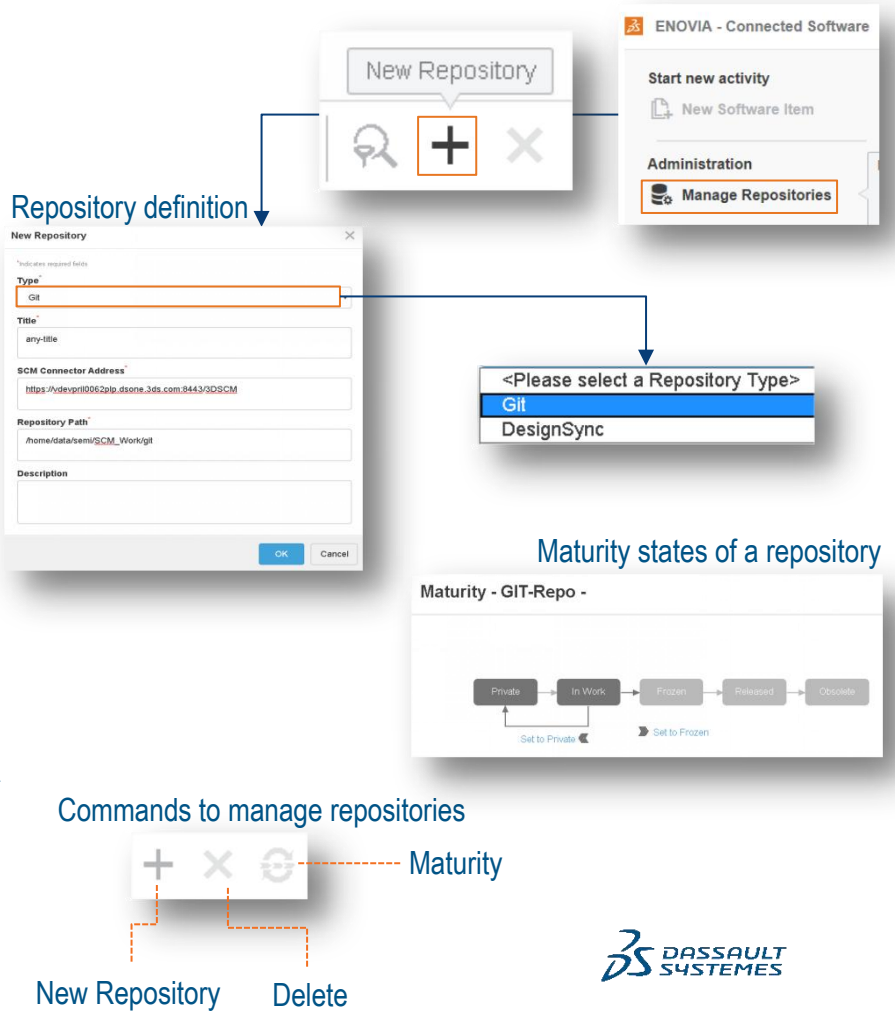


ENOVIA

DASSAULT SYSTEMES

软件库管理

- 软件库应该由拥有 "所有者 "角色的用户事先创建。.
- “管理存储库 ”部分包括以下命令：
  - ‘新资料库’
    - 创建一个新的资源库
  - ‘删除’
    - 不需要的资源库可以被删除。
  - ‘成熟度’
    - 促进或降低库的成熟度状态
    - 存储库只有在不处于 "私有 "成熟状态时才能使用。



注意：合作空间的 "所有者 "访问角色是必须的，以便能够创建新的资源库。

主要功能描述(1/2)

	亮点	能力	关键功能
1	获得对针对一个版本提交的最新软件内容的可见性，以改进代码审查、知识转移和根本原因分析。	**软件管理和可追溯性** 软件工程师可以在不影响日常开发工作的情况下，将软件代码和构建从外部SCM工具联合到3DEXPERIENCE平台，以便在多学科产品设计和组成的背景下进行治理和追踪。 **软件库** 一个软件库（外部）消除了对单一源代码管理（SCM）工具的限制，以实现协作和可追溯性。软件工程师可以将其生态系统中的外部源代码库连接到3DEXPERIENCE平台，以实现可追溯性和重复使用。 **软件item内容** 不离开3DEXPERIENCE平台就能充分了解软件实施的内容。可以查看外部资源库中存在的全部文件和文件夹结构.	Create Software Logical Items
2	进行高效和协作性的基于网络的代码审查	**软件内容审查:** 在3DEXPERIENCE平台内，软件工程师可以进行代码审查和共享代码，弥补了开发中环境和管理平台之间的差距。 **软件item的生命周期:** 定义和管理 软件工程项目的版本生命周期可以在云上定义和管理。软件实施的成熟度可以与它的成熟度和发布状态一起被治理。	Software Code Review • Real-time Access to the Software Code for review • Promote software logical item to Frozen
3	改善多学科协作	**多学科的产品设计** 3DEXPERIENCE平台上的 "连接软件 "应用允许工程人员在单一平台上管理软件及其组件和硬件。软件工程师的角色允许连接到外部SCM，提供对软件开发环境的可追溯性；捕捉软件代码和构建，在多学科产品的背景下实施。	1. Create Software Physical Product 2. Integrate Software into Multi-discipline Structure • Insert into existing product structure • Real-time Access to the Software Build for review

ENOVIA

DASSAULT SYSTEMES

主要功能描述(2/2)

	亮点	能力	关键功能
4	通过纳入企业治理和流程，包括任务、变更和需求管理，执行最佳实践。	**联合源代码管理** 软件工程师可以连接到外部SCM资源库，发布软件内容元数据或将软件内容导入到3DEXPERIENCE平台，以建立从软件设计和架构到开发的可追溯性。 软件工程师可以在整体产品定义中管理软件成熟度。 此外，软件工程师可以发布元数据或将软件构建内容导入到3DEXPERIENCE平台，以实现软件代码和构建工件之间的全面追踪。	1. Release the Software Items • Promote software physical product to Frozen • Promote to Released state 2. Raise Issues or Changes • Raise issues on Software Physical Products • Raise Change Actions on Software Items/Physical Products
5	确保互操作性和可追溯性	**软件版本控制** 每个修订版都可以创建新的软件版本。 版本生命周期可以独立于其他版本进行管理，并保持对每个新版本软件的历史的完全可追溯性。	Revisions and Branches

Integrated Software Development Process

- 管理软件库
  在网络上创建软件逻辑项目
  连接到外部资源库

- 开始软件代码开发（在外部SCM工具中）。

- 进行基于网络的代码审查

- 生成软件构建和执行测试（在外部SCM工具中）。

- 将软件逻辑项目提升到冻结状态
  创建新的软件物理产品，并将其与外部SCM工具中的构建相连接

- 将软件物流产品插入现有的硬件产品结构中
  进行基于网络的代码审查，以及对软件和硬件的整合的审查
  促进软件物理产品的冻结

- 发布软件逻辑item以及软件物理产品
- 使用变更程序或
  手动释放

- 为软件项目或实物产品的修改提出问题或启动修改程序
- 一个软件项目/实物产品在其生命周期内的演变是通过以下方式管理的
  - 修订和/或
    分支机构

注意：实际的软件代码开发、构建和测试是在外部SCM工具中完成的。

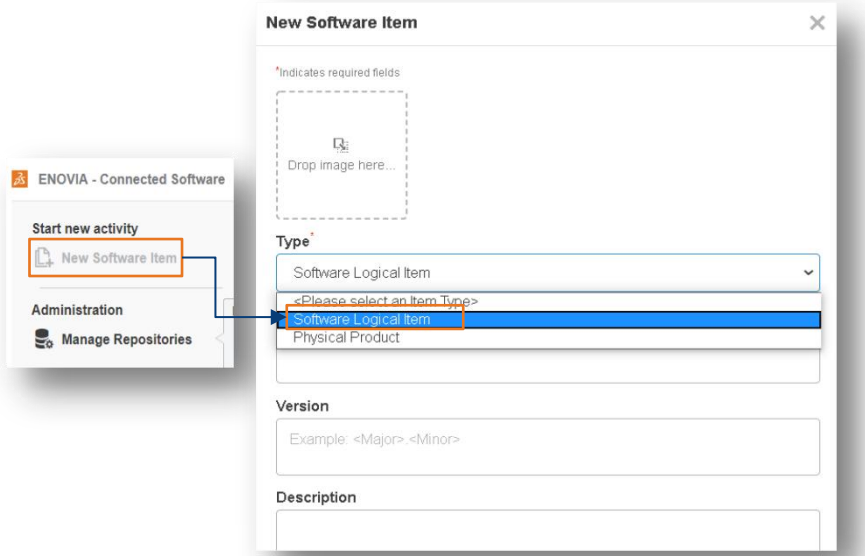
ENOVIA

DASSAULT SYSTEMES

软件管理和可追溯性

创建软件逻辑item

- 软件逻辑项的创建是为了开始新的软件开发。
- 软件代码在外部单片机系统中开发，通过在平台中创建的存储库将软件项目与外部单片机系统连接起来，从而在平台中进行访问。
- 这种与外部SCM系统的连接是通过为每个软件项目定义以下参数建立的
  - 存储库
    - 从平台中的用户定义的存储库列表中选择
  - Item地址
    - 外部SCM资源库的路径和名称
  - 版本地址
    - 来自外部SCM资源库的分支标识符
    - 在DesignSync中，它被称为 "选择器 "或 "版本标识"
    - 在GIT中，它被称为 "提交ID "或 "分支名称'



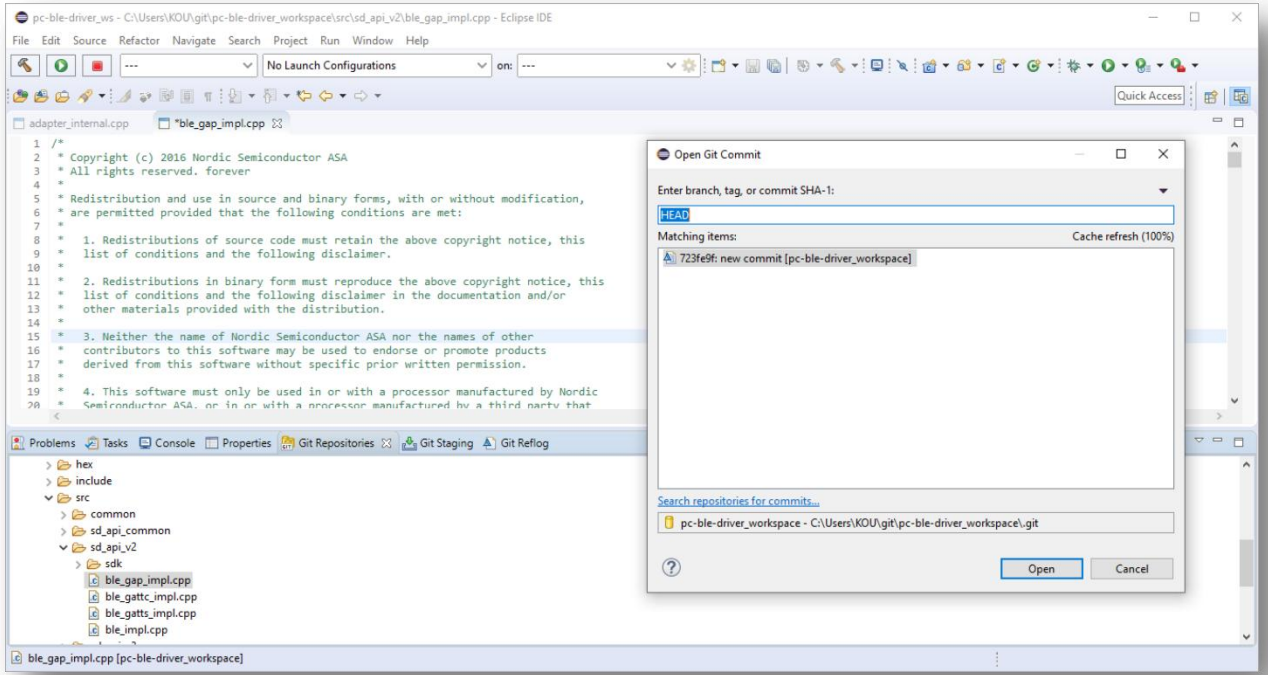
Parameters of repository connection

ENOVIA

DASSAULT SYSTEMES

在SCM系统中开发代码

- 软件代码开发是在外部SCM系统中进行的，如：
  - GIT
  - DesignSync

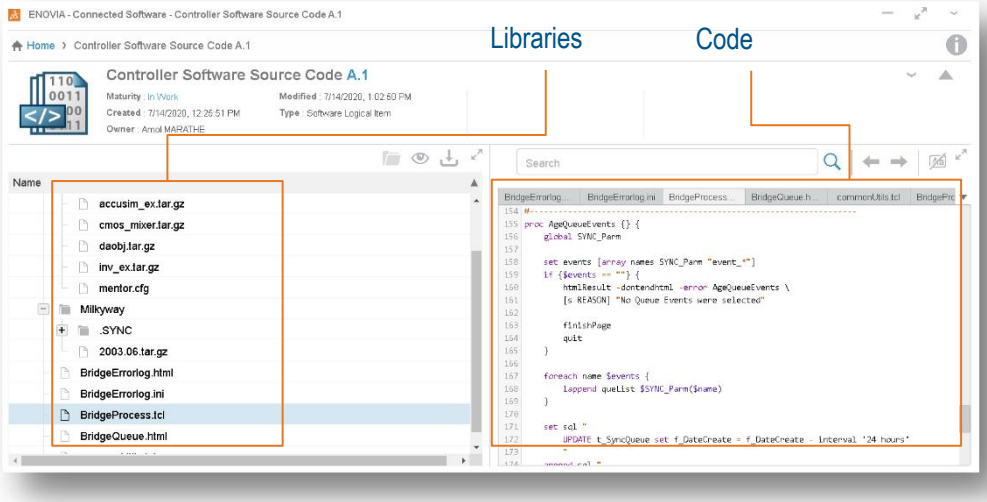


软件内容审查

软件代码审查 - 实时访问软件代码进行审查

- 软件逻辑项被打开以显示其内容
  - 库
    - 外部资源库中内容的文件和文件夹结构
  - 代码
    - 在语言敏感的查看器中显示的软件源代码
- 在代码查看器中，你可以对代码实时执行以下操作：
  - 代码审查
  - 搜索
  - 下载

Software Logical Item - *Source code access*



ENOVIA

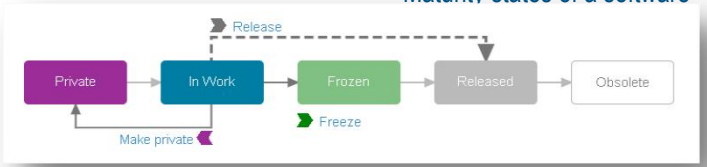
DASSAULT SYSTEMES

软件内容审查

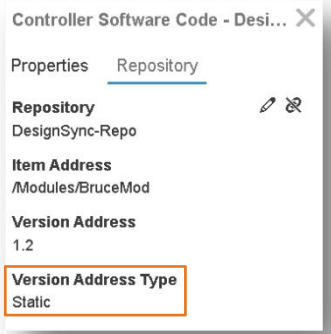
软件代码审查--促进软件逻辑项目的冻结

- 软件逻辑项目被提升到冻结状态，以冻结进一步的修改
- 注意，要改变一个软件项目的成熟度状态，它必须与外部存储库的静态（非动态）版本地址相连接
  - 例如，在GIT中，提交ID代表静态版本地址，而分支名称代表动态版本地址

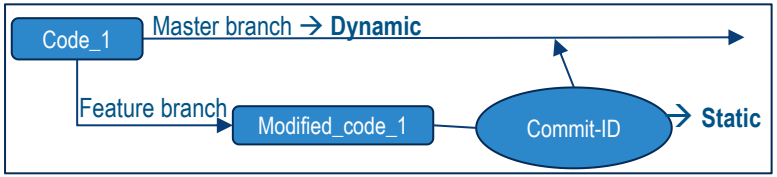
Maturity states of a software



Static version address in DesignSync



Static version address in GIT



ENOVIA

DASSAULT SYSTEMES

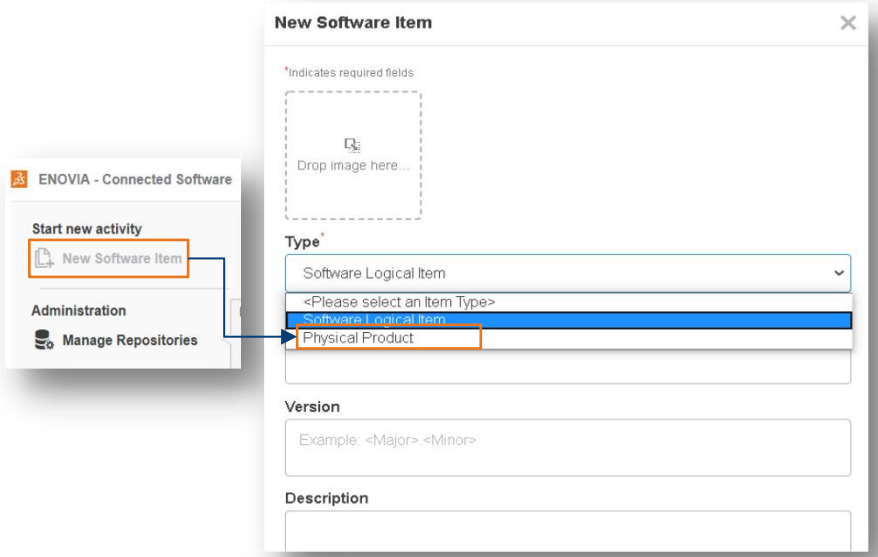
构建和测试软件代码

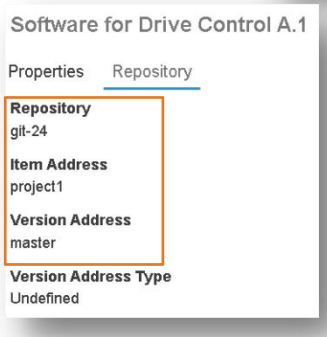
- 在外部SCM工具中生成软件构建并执行测试，如：
  - GIT
  - DesignSync

多学科的产品设计

创建软件物理产品

- 软件物理产品的创建是为了代表软件的物理形式，如软件构建。
- 软件构建和测试在外部SCM系统中进行，通过在平台中创建的存储库将软件项目连接到外部SCM系统，从而在平台中访问构建工件。
- 这种与外部SCM系统的连接是通过为每个软件项目定义以下参数来建立的.
  - 存储库
    - 从平台中的用户定义的存储库列表中选择
  - Item地址
    - 外部SCM资源库的路径和名称
  - 版本地址
    - 来自外部SCM资源库的分支标识符
    - 在DesignSync中，它被称为 "选择器 "或 "版本ID"
    - 在GIT中，它被称为 "提交ID "或 "分支名称"。





ENOVIA

DASSAULT SYSTEMES

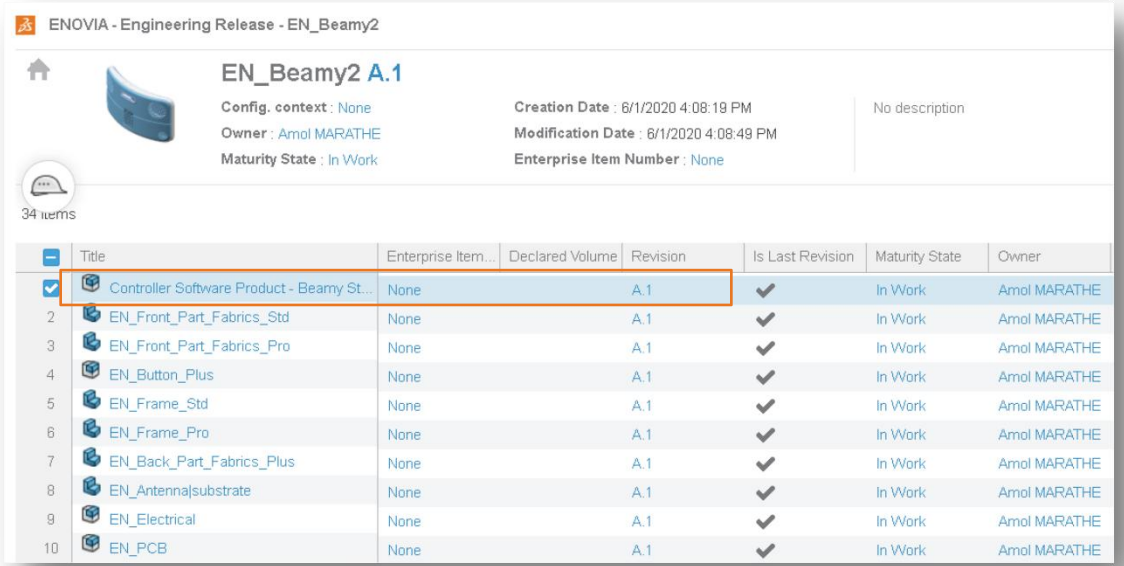
多学科的产品设计

将软件整合到多学科的结构中（1/2）。

插入到现有的产品结构中

- 连接的软件应用允许工程在一个单一的平台中与硬件一起管理软件及其组件
- 通过在现有的硬件产品结构中插入软件物理产品来准备一个多学科的架构

Multi-discipline Architecture



NOTE: Preparation of multi-discipline architecture requires additional roles such as 3D Product Architect (PAU) or Product Release Engineer (XEN) to edit the hardware product structure

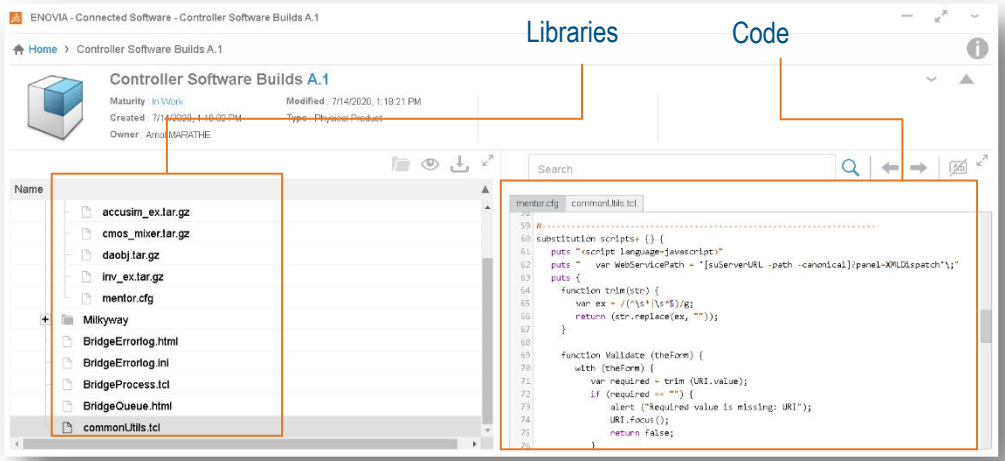
多学科的产品设计

将软件整合到多学科的结构中（2/2）。

实时访问软件建设，以便审查

- 软件物理产品被打开以显示其内容
  - 库
    - 外部资源库中内容的文件和文件夹结构
  - 代码
    - 在语言敏感的浏览器中显示的软件构建（或编译的可执行文件）.
- 在代码查看器中，你可以对代码实时执行以下操作：
  - 代码审查
  - 搜索
  - 下载

Software Physical Product – *Software builds access.*



ENOVIA

DASSAULT SYSTEMES

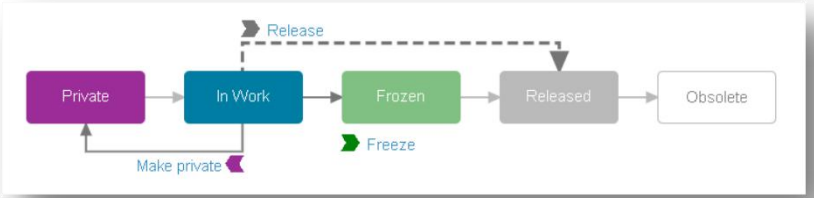
整合源代码管理

发布软件项item(1/2)

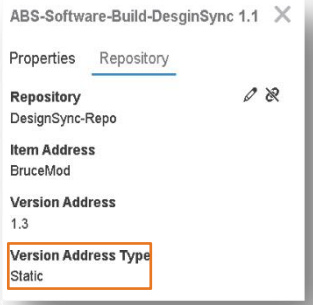
提升软件物理产品到“冻结”状态:

- 审查后，软件物理产品被提升到冻结，以冻结进一步的修改
- 注意，要改变一个软件项目的成熟度状态，它必须与外部存储库的静态（非动态）版本地址相连接
  - 例如，在GIT中，提交ID代表静态版本地址，而分支名称代表动态版本地址

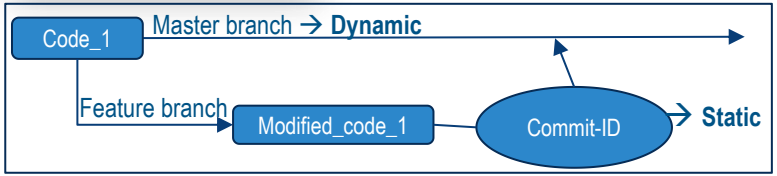
Maturity states of a software item



Static version address in DesignSync



Static version address in GIT



ENOVIA

DASSAULT SYSTEMES

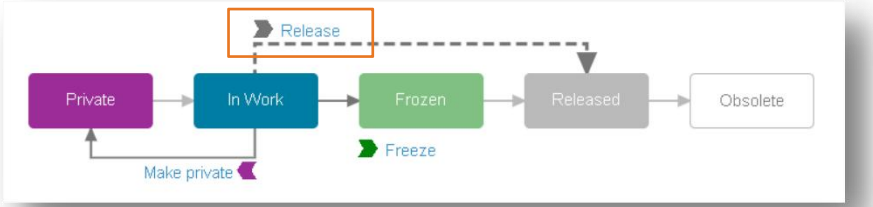
整合源代码管理

发布软件项item(2/2)

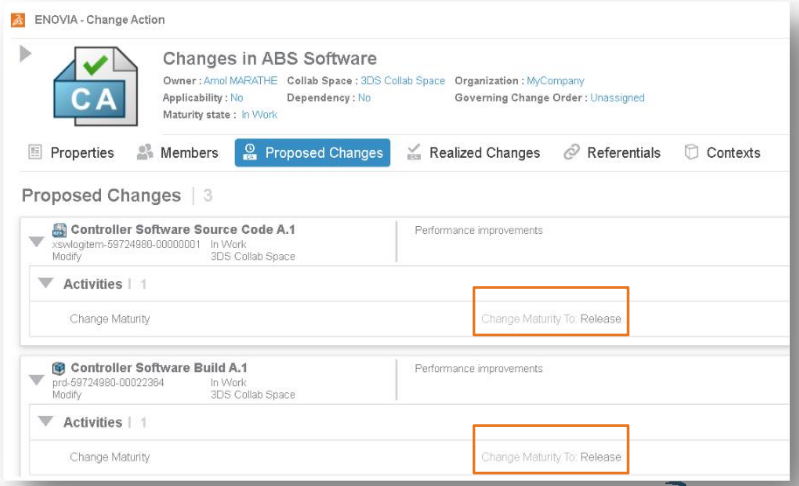
提升为释放状态：

- 在多学科架构内成功实施后，软件逻辑项目以及软件物理产品被提升到发布状态。：
  - 使用变更程序或
  - 手动发布

Manual release of software item



Automatic release of software item using Change Action



ENOVIA

DASSAULT SYSTEMES

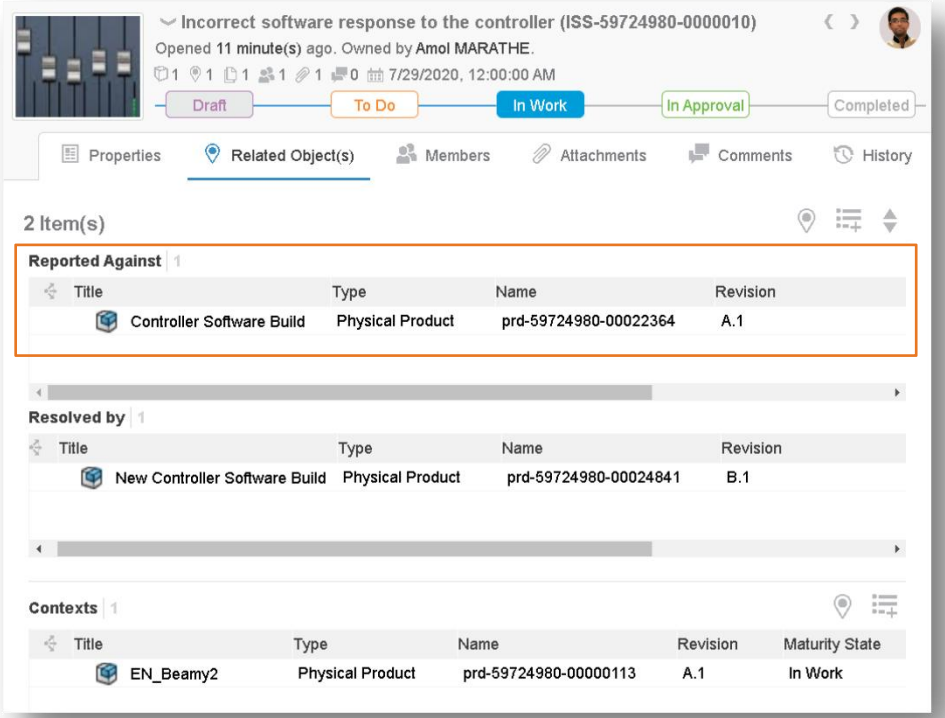
源代码管理

提出问题或改变行动(1/3).

提出关于软件物理产品的问题

- 通过添加上下文信息，针对有问题的软件物理产品报告 "问题"，并分配给负责任的人和组织
- 它允许提交、跟踪、优先处理和分配软件问题，使用快速解决流程来推动效率
- 跨组织的团队合作解决软件问题，并以正确的信息或正式变革过程的开始来结束它

Issue reported against a faulty software physical product



ENOVIA

DASSAULT SYSTEMES

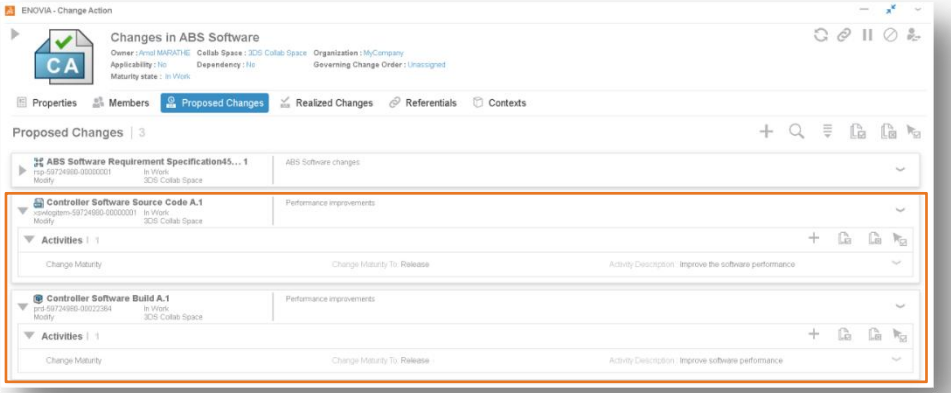
源代码管理

提出问题或改变行动(2/3).

提出对软件逻辑项目/物理产品的更改行动：

- 变更行动用于正式管理软件代码的变更过程
- 它为用户提供了一个受控的、明确定义的过程，以要求对软件项目进行更改，管理更改的批准，并跟踪更改的实施
- 参与变革过程的每个人都被分配了责任，决定了他们在这个过程中可以做什么，需要做什么
- 为软件项目创建的 "变更行动 "将跟踪所有的修改和它们在 "已实现的变更 "下的批准
- 更改行动也可以在软件实物产品上报告，以更改实物产品中的附加文件/文件

Software Item and Physical Product as Proposed Changes of Change Action



ENOVIA

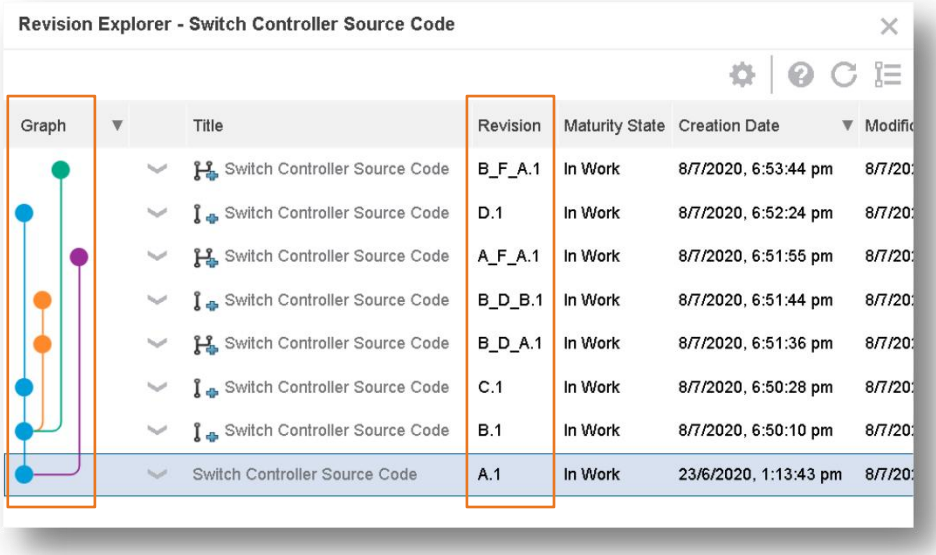
DASSAULT SYSTEMES

软件版本管理

修订和分支(3/3).

- 一个软件逻辑项目/物理产品在其生命周期内的演变是通过以下方式管理的
  - 修订版
    - 为了在其生命周期内捕获一个特定的衍生产品
  - 分支
    - 从现有的内容创建新的内容，保持与原始内容的链接
- 可对软件修订进行管理
  - 无变更流程:
    - 创建新的软件逻辑项目和软件物理产品的修订版，以启动新的开发
    - 发布新的版本
  - 有变更流程:
    - 新的修订版被自动创建和发布

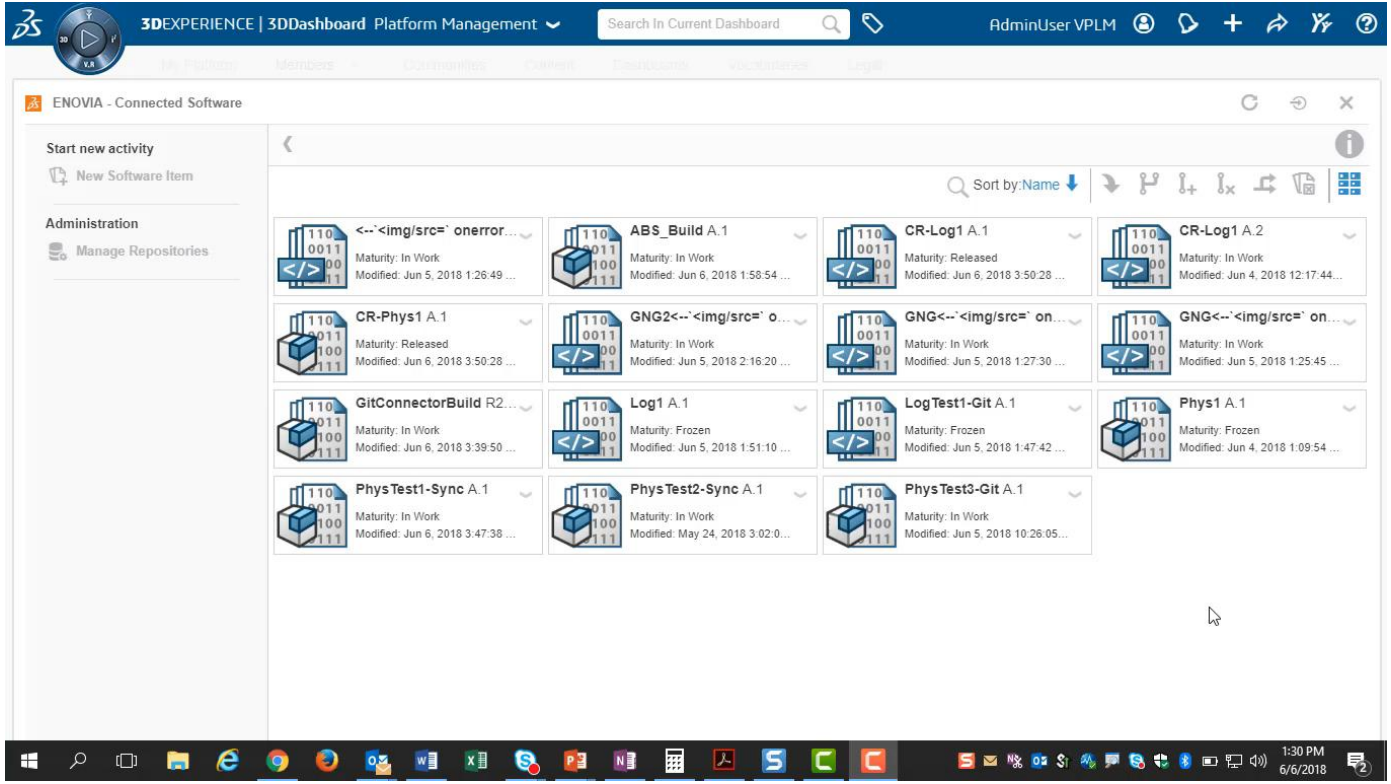
Revisions and Branches



ENOVIA

DASSAULT SYSTEMES

应用示例



ENOVIA

DASSAULT SYSTEMES