

Hyperledger Fabric v2.0 更新说明

v1.0 是 Hyperledger Fabric 的第一个主版本，Fabric v2.0 为用户和运营商提供了重要的新特性，包括对新应用程序和隐私模式的支持，增强了对智能合约的管理和对节点操作的新选项。

在 v2.0 上的每一个 v2.x 次要发行版本都有少量特性更新、改进和 bug 修复。

v2.2 是 Fabric v2.x 的第一个 LTS（long-term support，长期支持版本）。后续将在 v2.2.x 版本中提供修复，直到发布新的 LTS 版本。

下面我们就来看看 Fabric v2.0 发布的一些亮点……

智能合约的去中心化治理

Fabric v2.0 引入了智能合约的去中心化治理，它为您在 Peer 节点上安装和启动链码提供了一个新的流程。新的 Fabric 链码生命周期支持多个组织在链码和账本交互之前协商链码的参数，例如链码背书策略。和以前的生命周期相比，新的模式有几个改进：

- **多个组织必须认同链码参数** 在 Fabric 1.x 版本中，一个组织可以为通道中的所有其他成员设置链码的参数（例如背书策略），这些成员只能拒绝安装链码，因此不能参与和该链码相关的交易。新的 Fabric 链码生命周期更加灵活，它既支持中心化的信任模型（如以前的生命周期模型），也支持去中心化的模型，去中心化的模型要求在链码在通道上变为活动状态之前，要有足够数量的组织就背书策略和其他细节达成一致意见。
- **更周密的链码升级过程** 在以前的链码生命周期中，升级交易可能由单个组织发起，这会给尚未安装新链码的通道成员带来风险。新的模式要求只有在足够数量的组织批准升级后才能升级链码。
- **更简单的背书策略和私有数据集合更新** Fabric 生命周期支持在不重新打包或重新安装链码的情况下，更改背书策略或私有数据集合配置。用户还可以利用新的默认背书策略，该策略要求获得通道上大多数组织的背书。在通道中添加或删除组织时，会自动更新此策略。
- **可查验的链码包** Fabric 生命周期将链码封装在可读性更强的 tar 文件中。这使得检查链码包和跨多个组织协调安装变得更加容易。
- **使用一个包在通道上启动多个链码** 以前的生命周期在安装链码包时，会使用打包时指定的名称和版本定义通道上的链码。您现在可以使用同一个链码包，以不同的名称在同一通道或不同通道上多次部署它。例如，如果您想在链码的“副本”中跟踪不同类型的资产。
- **通道成员之间的链码包不需要为同一个** 组织可以自己扩展链码，例如为了他们组织的利益执行不同的验证。只要有所需数量的组织的链码执行结果匹配并为交易背书，该交易就会被验证并提交到帐本中。这还允许组织按照自己的时间单独推出小的修补程序，而不需要整个网络同步进行。

使用新的链码生命周期

对于现有的 Fabric 部署，您可以继续使用 Fabric v2.0 之前的链码生命周期。新的链码生命周期只有在通道应用程序功能更新到 v2.0 时才会生效。关于新的链码生命周期完整细节教程请参考 [chaincode4noah](#)

用于协作和共识的新链码应用程序模式

支持新的链码生命周期管理的相同的去中心化的达成协议的方法也可以用于您自己的链码应用程序中，以确保组织在数据交易被提交到帐本之前同意它们。

- **自动检查** 如上所述，组织可以在链码功能中添加自动检查，以便在背书交易提案之前验证附加信息。
- **去中心化协议** 人们的决定可以通过链码中的多个交易来建模。链码可能要满足来自不同组织的参与者在账本交易中的协议和条件。然后，最终的链码提案可以验证所有交易者的条件是否得到满足，并最终“解决”所有通道成员之间业务交易。有关私有条件的具体示例，请参见资产转移场景文档 [私有数据](#)。

私有数据增强

Fabric v2.0 还启用了使用和共享私有数据的新模式，不需要为所有想要进行交易的通道成员组合创建私有数据集。具体地说，您不是在多个成员的集合中共享私有数据，而是可能想要跨集合共享私有数据，其中每个集合可能包括单个组织，也可能是带有一个监管者或审计师的组织。

Fabric v2.0 中的几个增强使得这些新的私有数据模式成为可能：

- **共享和验证私有数据** 当私有数据与不是集合成员的通道成员共享时，或者与包含一个或多个通道成员的另一个私有数据集共享时（通过向该集合写入密钥），接收方可以利用链码的 `GetPrivateDataHash()` API 验证私有数据是否与以前交易中创建的私有数据在链上哈希相匹配。
- **集合级别的背书策略** 现在可以选择使用背书策略来定义私有数据集，该背书策略会覆盖集合内键的链码级的背书策略。该特性可用于限制哪些组织可以将数据写入集合，并且正是它启用了前面提到的新的链码生命周期和链码应用程序模式。例如，您可能有一个链码背书策略，该策略要求大多数组织背书，但对于任何给定的交易，您可能需要两个交易处理组织在它们自己的私有数据集中单独背书它们的协议。
- **每个组织的隐式集合** 如果您想利用每个组织的私有数据模式，那么在 Fabric v2.0 中部署链码时甚至不需要定义集合。不需要任何前期定义就可以使用隐式的特定组织集合。

了解更多关于新的私有数据模式请看 [私有数据](#)（概念文档）。更多关于私有数据集配置和隐式集合请看 [私有数据](#)（参考文档）。

外部链码启动器

外部链码启动器功能使运营者能够使用他们选择的技术构建和启动链码。默认构建和运行链码的方式与之前的版本相同，都是使用 Docker API，但是使用外部构建器和启动器就不必这样了。

- **消除 Docker 守护进程依赖** Fabric 以前的版本要求 Peer 节点能够访问 Docker 守护进程，以便构建和启动链码，但是 Peer 节点进程所需的特权在生产环境中可能是不合适的。
- **容器的替代品** 不再要求链码在 Docker 容器中运行，可以在运营者选择的环境（包括容器）中执行。
- **可执行的外部构建器** 操作员可以提供一组可执行的外部构建器，以覆盖 Peer 节点构建和启动链码方式。
- **作为外部服务的链码** 传统上，链码由 Peer 节点启动，然后连接回 Peer 节点。现在可以将链码作为外部服务运行，例如在 Kubernetes pod 中，Peer 节点可以连接到该 pod，并利用该 pod 执行链码。了解更多信息请查看 [将链码作为外部服务](#)。

了解更多关于外部链码启动功能请查看 [外部构建器和启动器](#)。

用于提高 CouchDB 性能的状态数据库缓存

- 在使用外部 CouchDB 状态数据库时，背书和验证阶段的读取延迟历来是性能瓶颈。
- 在 Fabric v2.0 中，用快速的本地缓存读取取代了 Peer 节点中那些耗费资源的查找操作。可以使用 core.yaml 文件中的属性 `cachesize` 来配置缓存大小。

基于 Alpine 的 docker 镜像

从 v2.0 开始，Hyperledger Fabric Docker 镜像将使用 Alpine Linux 作为基础镜像，这是一个面向安全的轻量级 Linux 发行版。这意味着现在的 Docker 镜像要小得多，这就提供了更快的下载和启动时间，以及占用主机系统上更少的磁盘空间。Alpine Linux 的设计从一开始就考虑到了安全性，Alpine 发行版的最小化特性大大降低了安全漏洞的风险。

示例测试网络

fabric-samples 仓库现在包括一个新的 Fabric 测试网络。测试网络被构建为模块化的和用户友好的示例 Fabric 网络，这使测试您的应用程序和智能合约变得容易。除了 cryptogen 之外，该网络还支持使用 CA (Certificate Authorities) 部署网络。

了解更多关于这个网络的信息，请查看 [使用Fabric的测试网络](#)。

升级到 Fabric v2.0

一个主版本的新发布带来了一些额外的升级注意事项。不过请放心，我们支持从 v1.4.x 到 v2.0 的滚动升级，因此可以每次升级一个网络组件而不会停机。

我们扩展和修改了升级文档，现在在文档中有了独立的主页 [升级到最新版本](#)。这里您将会发现文档 [Upgrading your components](#) 和 [Updating the capability level of a channel](#)，以及对升级到 v2.0 的注意事项的具体了解，[Considerations for getting to v2.0](#)。

发行说明

版本说明为迁移到新版本的用户提供了更多细节。可以具体地看一看新的 Fabric v2.0 版本中变动和废弃的内容。

- [Fabric v2.0.0 发行说明](#).
- [Fabric v2.0.1 发行说明](#).
- [Fabric v2.1.0 发行说明](#).
- [Fabric v2.1.1 发行说明](#).
- [Fabric v2.2.0 发行说明](#).

