**6.1 数字化转型的技术蓝图**

企业数字化转型要经历三个阶段，分别是信息化、数字化和智能化阶段。

信息化阶段，是企业数字化转型的起点。在这个阶段，企业将传统的纸质流程转化为电子化的流程管理。主要关注的是生产制造、物料转移、事务处理、资金流动和客户交互等流程的电子化。目标是提升企业流程管理的效率，实现无纸化办公。然而，在信息化阶段，企业的系统往往是离散的，数据也呈碎片化的状态。因此，信息化阶段的重点是构建一个高效的流程管理体系，以提高企业的运营效率和响应能力。

数字化阶段，是重塑商业模式和优化业务管理。当企业的信息化达到一定程度之后，随着业务的快速发展，原有的流程和系统已经无法满足企业的管理需求。此时，企业开始转向数字化阶段，从流程管理转向业务管理。企业对业务进行细粒度的拆分、分析和优化，以便更好地管理、分析和改善制造流程、业务流程和用户旅程等。数字化转型的核心在于数字对商业的重塑，这个过程通常伴随着组织结构的调整，以赋能企业的商业模式创新和突破。在这一阶段，企业信息化呈现出系统平台化、数据集约化和模型化的特点。

智能化阶段，是利用人工智能提升效率和创新力。随着企业拥有大量数据和人工智能技术的快速发展，智能化成为企业数字化转型的必然趋势。在智能化阶段，企业利用人工智能算法和模型，从数据中高效地提取业务知识。各种系统和应用变得越来越智能，能够自学习知识并创造新的知识。智能化的核心思想，是利用人工智能算法和模型解放生产力，寻找新的商机。此阶段企业的主要特点是系统自动化、数据模型化和智能化。

企业数字化转型是一个持续演进的过程，每个阶段都具有不同的重点和挑战。在这个数字化时代，企业必须借助AI科技，积极推动数字化转型，才能在激烈的市场竞争中立于不败之地。通过理解信息化、数字化和智能化的优势和应用场景，企业可以更好地规划和执行数字化转型的技术策略，支撑好营销、运营和产品策略。

企业数字化转型不仅是时代发展的要求，也是企业持续发展的关键。通过信息化、数字化和智能化的阶段性转变，企业能够提升运营效率、优化业务流程、加强数据管理和分析能力，从而实现商业模式的创新和突破。

下面，我们来阐述企业架构和云平台的架构设计。

1. 企业架构

架构是指在软件工程中，确定一种结构、组件、接口和关系的过程。它是一个基础框架，决定了内部各个部分之间的关系和交互方式。架构通常由一组设计原则、模式和约束条件组成，旨在帮助开发人员创建高质量、可靠且易于维护的软件。

从技术战略视角看，架构可以分为三种类型：软件架构、硬件架构和企业架构。软件架构是指软件的整体结构，包括各个组件之间的关系以及整体运行方式；硬件架构是指计算机中各个硬件设备之间的关系；企业架构则是指企业内部各个部门之间的关系，以及企业与外部环境之间的关系。

从业务战略视角看，架构类型分为业务架构、数据架构、应用架构和技术架构。业务架构聚焦在业务战略治理组织和关键业务流程上。数据架构关注组织的各类逻辑和物流数据资产，以及数据管理资源的结构。应用架构着重描述被部署的单个应用系统之间的交互，以及它们与组织核心业务流程之间关系的蓝图。技术架构是支持业务数据和应用架构的实施的基础，它包括必要的逻辑、软硬件能力。具体而言，技术架构涵盖了IT基础设施、中间件、网络通信以及部署处理的相关标准等。

数字化转型愿景是5至10年的战略规划，而企业架构蓝图是衔接战略与产品实施的桥梁，因此通过分层分级进行系统化的顶层设计，是产品和业务达成共识的路线图。因此，架构蓝图需要体现业务与数字技术的双向驱动：一方面，从业务和场景的思维入手，匹配数据和系统；另一方面，通过数据和技术来赋能业务，提升用户体验和运营效率。

Gartner对企业架构的定义是：将企业战略转化为企业变革的需求、原则及蓝图，并通过持续提升流程和管控流程来推动企业变革，促进企业战略的实现。

企业架构的目的是形成一个连贯而全面的组织蓝图，它具有长远视野，描绘了整个组织的全景，该蓝图与众多要素相关联。同时，企业架构也是一种路线图，可以指导组织逐步规划和维护业务目标，并通过技术来支持它们。

企业框架在研发阶段的主要工作及设计要素如下。

（1）在需求分析阶段，软件架构设计主要负责梳理非功能性系统需求，比如软件的高可维护性、高性能、高复用性、高可靠性、有效性和可测试性等。另外，架构设计还要定期分析客户不断变化的核心需求，严格把关和确核开发团队所提出的设计。

（2）在总体设计阶段，架构师的关注点是开发团队的技术能力和开发模式；AI技术引领业务设计是第一思维，能借助主流的云平台的成熟产品，引入中间件和核心组件，穿透业务，实现降本增效。

（3）在软件概要和详细设计阶段，架构师负责对整个软件体系结构、关键构件、接口和开发政策的设计。

（4）在代码编码阶段，架构师成为详细设计者和代码编写者的老师，并且要经常性地组织一些技术研讨会、技术培训班等来提升团队的技术能力。

（5）在软件测试交付阶段，架构师跟踪关注性能需求，同时开始为下一版本的产品是否应该增加新的功能模块进行决策。

企业框架在研发阶段，一是要保证服务之间高内聚、低耦合，二是要遵守可读性、可扩展、可用性和可维护性等软件设计原则，如表6-1所示。

表6-1　企业架构的设计原则

|  |  |  |
| --- | --- | --- |
| 原则名称 | 定义 | 优势分析 |
| 开闭原则 | 一个软件实体，如类、模块和函数，应该对扩展开放，对修改关闭；强调用抽象构建框架，用实现扩展细节 | 提高软件系统的可复用性和可维护性 |
| 依赖倒置原则 | 高层模块不应该依赖底层模块，二者都应该依赖其抽象；抽象不应该依赖细节，细节应该依赖抽象；针对接口编程，不要针对实现编程 | 可以减少类间的耦合性，提高系统稳定性，提高代码可读性和可维护性，可降低修改程序所造成的危险 |
| 单一职责原则 | 不要存在多于一个导致类变更的原因，一个类、接口、方法只负责一项职责 | 降低类的复杂度，提高类的可读性，提高系统的可维护性，降低变更引起的风险 |
| 接口隔离原则 | 用多个专门的接口，而不是使用单一的总接口，客户端不应该依赖它不需要的接口。注意：一个类对应一个类的依赖应该建立在最小的接口上；建立单一接口，不要建立庞大臃肿的接口；尽量细化接口，接口中的方法尽量少；注意适度原则 | 符合我们常说的“高内聚、低耦合”的设计思想，从而使类具有很好的可读性、可扩展性和可维护性 |
| 迪米特法则 | 一个对象应该对其他对象保持最少的了解，又叫最少知道原则；尽量降低类与类之间的耦合 | 降低类之间的耦合 |
| 里氏替换原则 | 如果对每一个类型为T1的对象A，都有类型为T2的对象B，使得以T1定义的所有程序P在所有的对象A都替换成B时，程序P的行为没有发生变化，那么类型T2是类型T1的子类。  引申意义：子类可以扩展父类的方法，但不能改变父类原有的功能 | 约束继承泛滥、开闭原则的一种体现；加强程序的健壮性，同时变更时也可以做到非常好的兼容性，提高程序的维护性、扩展性；降低需求变更时引入的风险 |
| 合成复用原则 | 尽量使用对象组合、聚合，而不是继承关系达到软件复用的目的。聚合has-a和组合contains-a | 可以使系统更加灵活，降低类与类之间的耦合度，一个类的变化对其他类造成的影响相对较少 |

2.云平台架构设计

数字化云平台建设，要解决三个核心问题：一是建设业务和数据中台，打破数据孤岛，确保数据共享和数据安全；二是实现数字化运营和决策，为一线市场提供最佳体验的产品和服务；三是借助云原生平台和微服务架构，提供可持续迭代的基础能力，为后续服务和产品扩展奠定坚实基础。

如图6-1所示，以华为数字云平台架构蓝图为例，平台架构蓝图共包括五层，各层级之间相互支撑、相互依赖和相互赋能。



图6-1　华为数字云平台架构蓝图

第一层是客户连接层。面向企业和消费者，聚焦在用户体验和开放能力上。

第二层是一线产品层。面向市场和客户的主业务层，支撑营销、交付、运维、渠道和销售的核心业务。该层级的产品设计关键是借助最先进的AI技术来增强用户体验和满意度。

第三层是业务中台层。布局了十朵垂直业务云，包括研发云、营销云、项目服务云、运维云、渠道云、供应链云、采购云、财务云、人力资源云、公共能力云。每一朵云都基于微服务设计数据和业务能力，向上可以灵活组合来支撑主业务层，向下按需调取数据底座的数据服务。在产品化之前，企业都在以项目为中心，并没有更多资源把能力沉淀成产品和平台。此时，很难形成资源共享和能力共用，一直出现“重复造轮子”的行为，形成烟囱式架构，造成研发资源浪费和数据能力内耗。

第四层是数据底座层。统一管理结构化和非结构化数据，打通数据供应通道，确保数据一致性和完整性，保障数据安全可控，这一层的核心是数据中台和数据仓库的建设。

第五层是IT平台。应用的构建、运行和管理都基于IT平台的云环境，应用充分利用和发挥云平台的弹性优势与分布式特点。上云的关键，是在利用云的能力和保护企业数据安全之间找到平衡点。一是企业在享受资源弹性和敏捷服务的同时，保证企业的核心数据在内网存储关联，更好地保证设计安全。二是通过混合云的模式，本地部署的核心服务和公用云的服务叠加，实现用户视角的一朵云。IT平台建设有以下几个关键要素。

（1）微服务架构。IT系统分为多个高内聚、低耦合、独立自治的服务，服务是采用无状态设计的，服务之间只能通过接口进行通信。

（2）弹性伸缩。服务可以基于规则和容量，按需水平伸缩。

（3）高可用。系统的任意服务实例失效，系统能快速发现、隔离并自动从故障中恢复，不影响系统整体的可用性。

（4）多租户。支撑多租户隔离，每个租户只能访问操作与自己相关的资源，不能访问、操作其他租户的任何资源。

（5）自动化运维。系统能够自动化部署、升级、扩容或者缩容，并能实现自动化监控、告警、故障定位和故障自愈等。