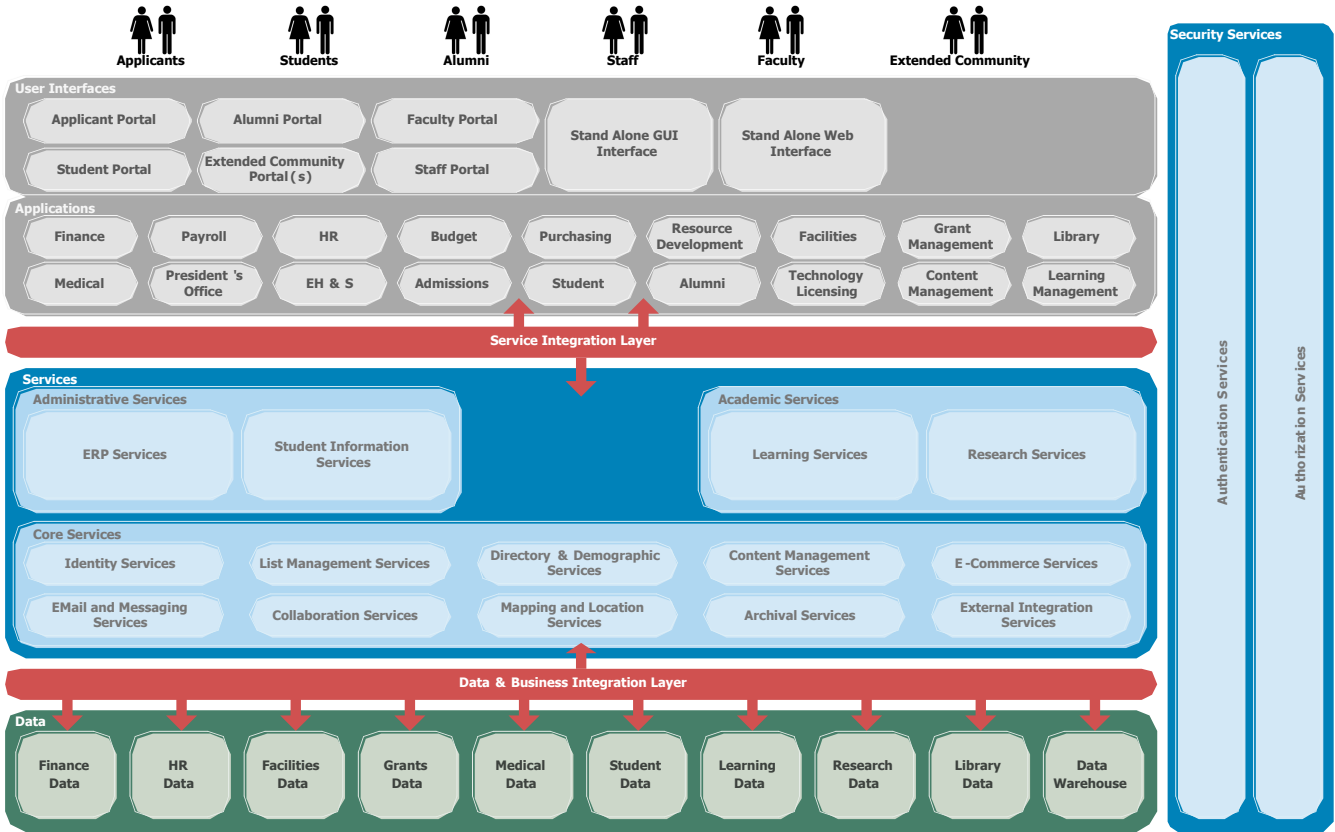




Logical Architecture Vision

The Logical Architecture Vision is a conceptual model evolving the enterprise architecture at MIT in the future. As such it raises as many questions as it answers, and provides a good forum for future discussions about the architecture.

- The **Data** layer illustrates the idea that enterprise data (not departmental data or departmental extensions to enterprise data) should be logically visible and consolidated by data domain. This does not imply that, for example, all student data must be physically located in the same database, or managed by the same system, but that there are a coherent set of rules for locating, a unified view of, and a standard way to access student data.
- The **Data & Business Integration Layer** speaks to the fact that there should be a consistent way of interacting with enterprise data, and a coherent strategy for sharing data across systems in the case that shadowing data is still necessary. This layer is key to achieving the goals of the data layer, i.e. logical consolidation and access to data by domain



- The **Services** layer represents the separation of re-usable services from application logic. The service layer will thus consist of services with clearly defined contracts that can be used by any application. The services have initially been classified in to three major groupings: Core Services for technical and basic services, Administrative for services that provide access to administrative data or processes and Academic Services for services specific to the education and research domains.
- The **Service Integration** layer is responsible for exposing the services in the architecture in a consistent manner while enabling services to be implemented in a variety of technologies. Ideally it should also define the standard contract for a service type, therefore allowing substitution of service implementation without affecting clients of the service.
- The **Applications** layer shows groupings of applications that are built for specific purposes. By leveraging services available in the architecture, applications should generally be quicker to develop and easier to maintain.
- The **User Interfaces** layer conveys the idea that users should have a single point of access for related functions that they use. This might be implemented as one or more portals.
- The **Security Services** layer, while conceptually similar to other types of services has been shown separately because it has significant impacts at all levels within the architecture. It will be necessary to apply access and control security to data, to services, to applications and finally to user interfaces.

