



## DXtera Technology Overview

*This technical overview serves to provide an initial introduction to select audiences. Please note the contents of this report are still in development and subject to change.*

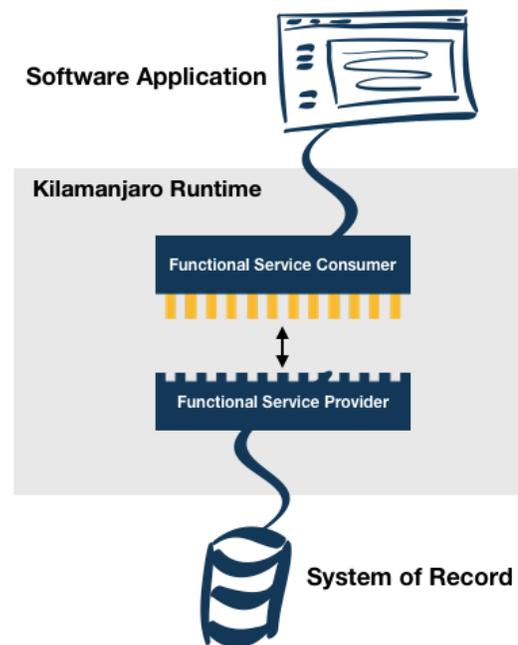
At the heart of most of DXtera's solutions and products is the Kilimanjaro runtime environment, an open-source framework that manages integration between software components that implement the [Open Service Interface Definition](#) (OSID) specifications. Kilimanjaro is a robust and flexible runtime that helps solve various kinds of integration and software development challenges.

The OSID specifications are a comprehensive, technology neutral and openly licensed collection of software service contracts that describe all common service and information models required by both legacy and next generation enterprise systems. Implementations of these contracts plug into the runtime environment to integrate and configure application access to existing or new software infrastructure.

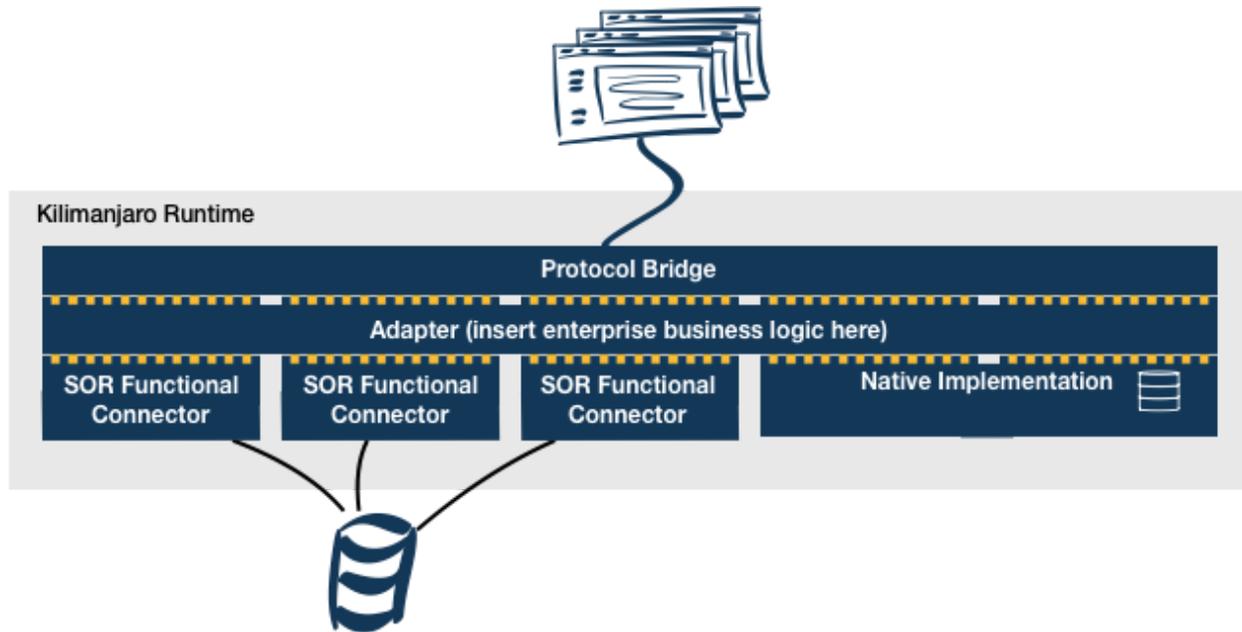
In DXtera's Integration Framework, Kilimanjaro is designed to facilitate secure, extensible and scalable real-time information exchange between educational enterprise systems and software applications that require access to the functionality and information provided by those systems. The framework is designed to allow organizations to deliver more direct and real-time information to their communities and customers. This is achieved through the development and deployment of reusable functional connectors that encapsulate the details of integration with a System of Record (SOR), like a student information system, learning system, human resource system, financial system, etc, and the needs of applications that require services.

In a component software framework, Kilimanjaro allows enterprise software developers to quickly design and build truly modular, plug and play systems utilizing reusable native implementations of the OSID specification. The framework helps to ensure separation of concerns between and among service components that provide operational functionality and application components or processes that support user requirements.

In addition to SOR implementations of the service contracts designed to integrate with existing or legacy enterprise systems, Native Implementations can be developed from scratch to provide



new infrastructure services. When deployed through the Runtime framework these services can be combined to implement “next generation”, modular and extensible enterprise infrastructure.



Various kinds of software can be developed to consume the specifications through the Runtime environment:

- User Applications - Next Generation user facing applications can be built to directly plug into underlying services as defined by the openly licensed OSID specifications through the runtime environment.
- ETL Applications - Software designed to extract, transform and load data from enterprise systems into operational data stores, data warehouses or other reporting and analytics systems can be streamlined through the framework. Extraction is accomplished through real-time integration. Transform is facilitated through the standardized use of the OSID models themselves.
- Protocol Bridges - Remote or mobile applications require network protocols like REST or WSDL to access enterprise information. Third-party application developers can securely consume data through these and other evolving protocol technologies (Protocol Buffers for example), standards-based protocols, or bespoke application protocols to meet the evolving needs of next generation applications.
- Business Logic Adapters - Software Adapters can be developed which both consume and implement the service contracts, and encapsulate various kinds of logic. These



allow insertion of business rules within the Framework itself, like authorization, system federation, and data validation, etc.

Some features of DXtera's Integration Framework:

**Modular.** Unlike other integration solutions, DXtera's integration framework is completely modular. Every component, from the runtime, to the business logic adapters, legacy connectors and native service implementations, can be updated, enhanced or even replaced as required. This is possible through ubiquitous use of the openly published service contracts at the core of its architectural design

**Extensible.** The technology landscape is ever-changing. Great ideas come and go. To make matters worse, institutional requirements are always evolving to meet the changing demands of faculty, students and communities. DXtera understands this, and has designed its framework to change and evolve with the times.

**Built for Education.** The DXtera integration framework is not just another ESB or technical integration platform. It is built to the business models of education institutions. Based on years of domain modeling with input from multiple institutions in multiple countries, DXtera is built on the most complete models of the educational enterprise in the industry.

**Member-driven.** Development of integration and infrastructure components is driven by the needs of DXtera's members, consisting of institutions, educational industry leaders and other related organizations that share similar information management and integration challenges. The member-owners of DXtera's software solutions help ensure that DXtera's offerings are designed for education, by education.

For more detailed information about DXtera's technology solutions strategy please go to DXtera's [Integration and Component Architecture](#) white paper.