Part III – TECHNICAL ARCHITECTURE
Chapter 4 – TECHNICAL SERVICES

CMS
CENTERS for MEDICARE & MEDICAID SERVICES

MITA
Medicaid Information
Technology Architecture

Table of Contents

PART III – TECHNICAL ARCHITECTURE 1
Chapter 4 – Technical Services 1
Introduction 3
Purpose 3
Scope 4
Technical Services Overview 4
Technical Service Areas 5
Access and Delivery 6
Intermediary and Interface 7
Integration and Utility 8
Technical Service Parts 9
Technical Service Development 11
Technical Service Solution Sets 12
Technical Services Flow 13
Benefits of Technical Services 16
Using Technical Services 17

List of Figures

Figure 4-1. Conceptual Relationship – Technical Function to Technical Service 4
Figure 4-2 TSAs Conceptual Diagram 6
Figure 4-3. Access and Delivery TSCs 7
Figure 4-4. Interface and Intermediary TSCs 8
Figure 4-5. Integration and Utility TSCs 9
Figure 4-6. Sample WSDL 12
Figure 4-7. Conceptual Relationship – Technical Function to Solution Sets 13
Figure 4-8. Service Orchestration Using the Conceptual Layout 14
Figure 4-9. Process Orchestration Modifications 14
Figure 4-10. SOA Composite Service 15
Figure 4-11. Federal SOA Framework 16

List of Tables

No table of figures entries found.

Introduction

This chapter provides a discussion of the Medicaid IT Architecture (MITA) Technical Services. Technical services provide the guidance and specifics to an IT staff (e.g., States or vendors) on how to design a MITA technical capability. The focus across the chapter is to delineate a technical service approach that:

- Focuses the technical assessment process on business needs.
- Provides a method for identifying technical services and applications available to move between capability levels.
- Determines the technical assets that are currently available in a state's infrastructure that can support future business needs.
- Reflects the technical needs, in conjunction with business service solution sets and the business service assessment that meet the business To-Be configuration.

This chapter defines the groupings of the Technical Service Area (TSA) and subgrouping of the Technical Service Classification (TSC). The definitions used in the Technical Capability Matrix (TCM) provide States with a technical capability assessment tool for the State Self-Assessment (SS-A).

The topics covered in this chapter include:

- Technical Services Overview
- Technical Service Areas
- Technical Service Parts
- Technical Service Development
- Technical Service Solutions Sets
- Technical Services Flow
- Using Technical Services
- Benefits of Technical Services

Purpose

The MITA Framework includes two categories of services: Business services and technical services. The TA derives the technical services from the business service requirements (as described in Part I, Chapter 3, Maturity Model), MITA principles, goals, and objectives, and the MITA TCM (as described in Part III, Chapter 7, Technical Capability Matrix). The TA discusses Technical services, which provide underlying technical functionality (e.g., forms management, security, etc.) in this chapter.

A TSC defines a standard and/or functionality for a technical process that aligns the common factors of a state's design with the Medicaid Enterprise. The technical service, as does a business service, allows two things:

- **Modularity** – A new implementation replaces an individual service without affecting the rest of the enterprise. For example, an enterprise can replace a service that is currently a wrapped Common Business Oriented Language (COBOL) application with a Commercial Off-the-Shelf (COTS) product or Oracle Java Platform, Enterprise Edition (J2EE) C++ program without changing any of the external interfaces.
- **Interoperability** – With interoperability, services can exchange information without the need to perform integration activities. The TA can achieve this by consistently adhering to sound Service-Oriented Architecture (SOA) design principles and design standards. When these activities carry over to the standardization of services contracts and their underlying data models, the TA can achieve a base level of automated interoperability across services.

Scope

Independence is an important characteristic of a technical service. A different service can easily replace an independent service, provided the new service meets the needs of the user. Services should also be location independent because, in today's IT environment, a service does not have to collocate with the users of that service.

Service owners document how the service performs so other computers are able to use the service. The documentation includes the functions included in the service (e.g., expected output, error checking, accuracy, etc.) and describes how to obtain the service and how other computers may request the service. Technical services should be implementation-neutral. To achieve a high level of loose-coupling, these services should not specify platform, binding protocols, programming models, operating systems, underlying infrastructure technologies, or other execution details to deploy the service.

Technical Services Overview

MITA Technical Services are component-driven, technical outlines that leverage a common, standardized vocabulary to allow intrastate and interstate agencies, federal partners, and health care stakeholders to leverage collaboration. The TA organizes the structure in a hierarchical fashion from groupings of the TSA to subgroupings of the TSC. As described in Part III, Chapter 2, Business Services, and depicted in **Figure 4-1** below, technical services align a technical function (i.e., authentication, audit log, etc.) with a maturity level. This process is similar to business process aligning with a maturity level to form an associated business service. While the goal is that each technical service aligns with the defined MITA Framework Service Standards and/or proven and approved solution set, the MITA Framework does not require such alignment.

Figure 4-1. Conceptual Relationship – Technical Function to Technical Service

Technical services define what services a State Medicaid Enterprise should design for required generic support functions. The goal of the MITA Framework is to specify services that enable interoperable Medicaid services. An orchestration process weaves individual services and messages with other Business and technical services, as defined in the Part III, Chapter 5, Application Architecture.

Quantifiable measures for technology require adoption of standards (i.e., information and technical) and measuring the performance of the system. The State Medicaid Agency (SMA) may adopt a standard and use quantifiable metrics to measure compliancy. The MITA team revised the TCM to include TSA and TSC to evaluate the technical maturity of a system and provide a guide for improvement. See Part III, Chapter 7, Technical Capability Matrix, for more details.

The TCM focuses on technical services, Application Architecture (AA), Technical Standards, and common definitions for increasing a maturity level. For example, utility services or generic architecture elements such as SOA or API measure maturity. The Organization for the Advancement of Structured Information Standards (OASIS) publishes several standards for use by States. For example, the publication *OASIS Service-Oriented Architecture End-to-End Resource Planning (SOA-EERP)* provides methods for optimizing deployment of services onto a SOA.

The SMA uses the Strategic Performance Model to adapt and extend a core set of measures. As the organization changes, it introduces additional measures and generates a more complete view of daily operations. As a state reaches a certain level of achievement in technology, it identifies new opportunities for improvement. Their achievements result from a combination of process changes, new technologies, new skills, and the right selection of resources.

The MITA TA supports accurate and timely processing of claims (including claims of eligibility), adjudications, and effective communications with providers, beneficiaries, and the public. The Centers for Medicare & Medicaid Services (CMS) intends to provide additional guidance concerning performance standards – both functional and non-functional, and with respect to Service Level Agreements (SLA) and Key Performance Indicators (KPI). CMS expects to consult with States and stakeholders as we develop and refine these measures and associated targets. This list of measures focuses on very core elements/indicators of success; States should also consider adding state-specific measures to this list. Solutions should produce transaction data, reports, and performance information that would contribute to program evaluation, continuous improvement in business operations, and transparency and accountability.

Technical Service Areas

A technical service is a piece of software that executes a generic IT capability. It has a defined interface for its invocation, performs a defined function that corresponds to the capability, and returns defined results. The MITA Technical Services Framework consists of three (3) TSAs:

- **Access and Delivery** – Encompasses design drivers and enablers such as web browser connectivity, language support, Customer Relationship Management (CRM), and forms and reports services. The access and delivery functions have a direct impact on the state staff, the public, providers, and all other stakeholders. The span of coverage

of the services offered will tend to change over time as the demands and technology needs of the end-user evolve.

- **Intermediary and Interface** – Contains drivers and enablers, such as process orchestration, workflow and relationship management functionality. The Enterprise Service Bus (ESB) offerings handle the intermediary services (sometimes referred to as middleware). The interface services tie to connectivity offerings of the nearby Medicaid Enterprise entities and other external organizations that require a connection.
- **Integration and Utility** – Includes design drivers and enablers such as solution stacks, database access layer services, scalability, application versioning, and verification type utility services. These core service components will likely be a combination of the unique services and a set of reusable services across the Medicaid Enterprise.

Figure 4-2 shows the conceptual depiction for the MITA TSAs.

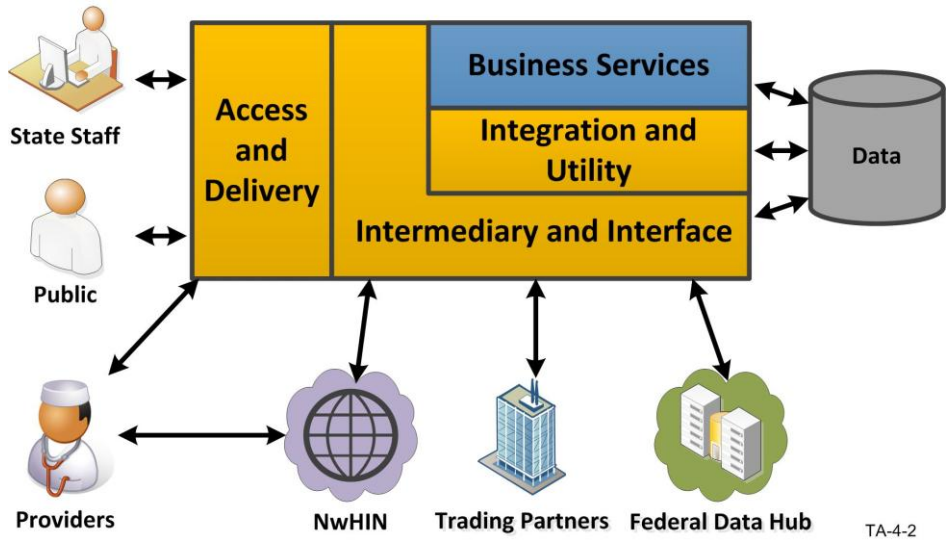

Figure 4-2 TSAs Conceptual Diagram

The layering of the types of services requires a combination of services for an end-user to retrieve data. The Interface and Integration TSA handles orchestration of events.

Access and Delivery

The Access and Delivery TSA contains five (5) TSCs. **Figure 4-3** depicts the TSCs and examples of potential solutions and/or standards. These technical services combine with Intermediary and Interface services along with Integration and Utility services in a composite manner to provide the full string of functionality required to satisfy the end-users needs. The need to leverage proven technology standards plays a major part in covering the varying devices used by the general public, state staff, and the provider community. There are needs to support a requesting and forwarding community, SLAs are especially important. Concise application tuning for influencing minimal throughput for these services should be part of the creation process of these application services.

A key portion of this TSA includes the usage of standardized security and privacy mechanisms. The protection of data assets from both an internal and external user base is of utmost concern on the Medicaid Enterprise, so it requires the usage of established role-based security functions. A role-based solution is a primary solution. However, using a layered manner to safeguard state assets to cover the various types of security and privacy needs is common. Additional security and privacy details, including standards information, are in Part III, Chapter 6, Technical Standards.

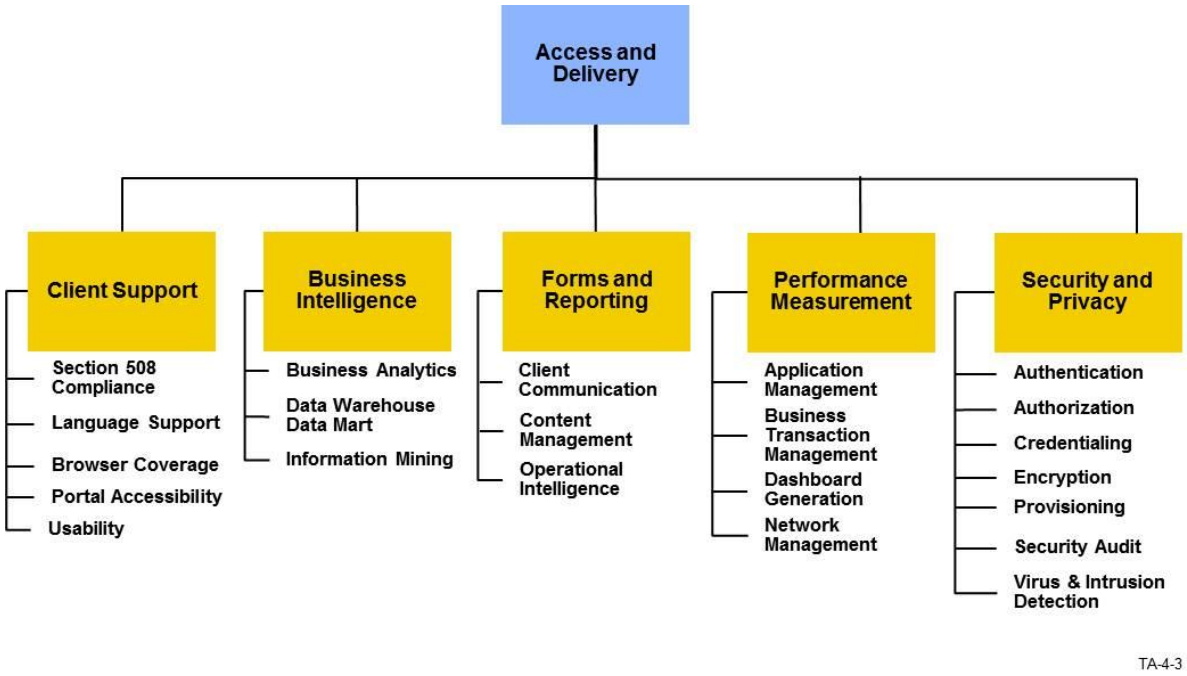

Figure 4-3. Access and Delivery TSCs

Intermediary and Interface

The Interface and Integration TSA contains six (6) TSCs. **Figure 4-4** depicts the TSCs and examples of potential solutions and/or standards. The interactive activities associated with the Medicaid Enterprise intermediary services require an affective middleware solution set. This TSA combines the generally predetermined interface activities with the transactional functions performed by the intermediary services. The ability to interface with the primary Medicaid Enterprise data stores requires a strong tie to security and privacy services.

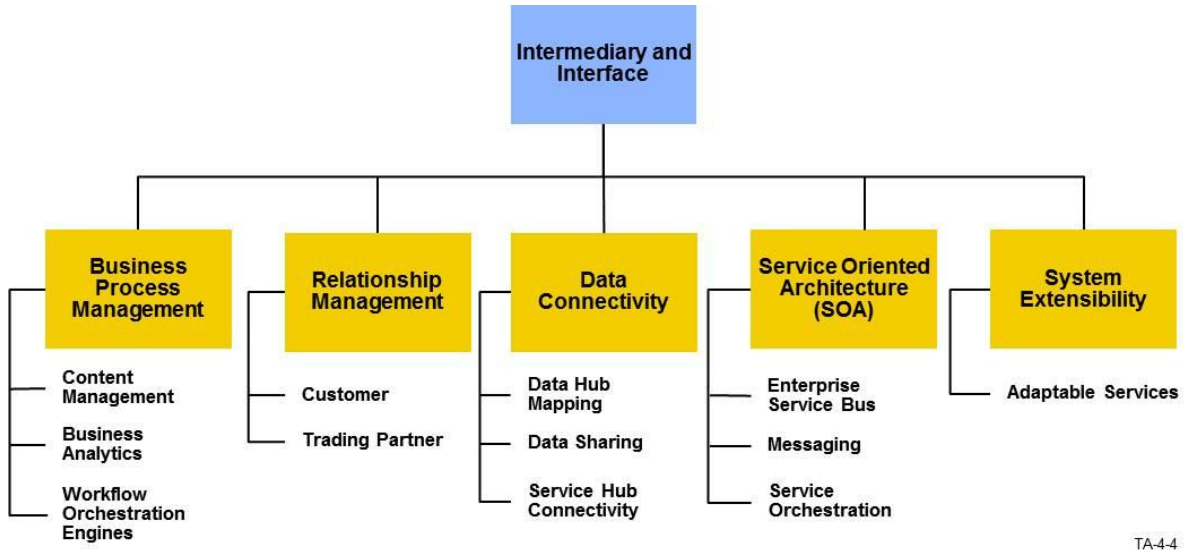

Figure 4-4. Interface and Intermediary TSCs

Integration and Utility

The Integration and Utility TSA contains six (6) TSCs. **Figure 4-5** depicts the TSCs and examples of potential solutions and/or standards. The core services that have a specific Medicaid function outside of the business services classify as utilities services. Other general functions, like logging and configuration management are part of the integration activities of this TSA. One of the critical internal activities to the Medicaid Enterprise involves the decision management events used by claims processing many of the other operational management processes. The Integration and Utility TSA classify these functions, which link closely to the business services.

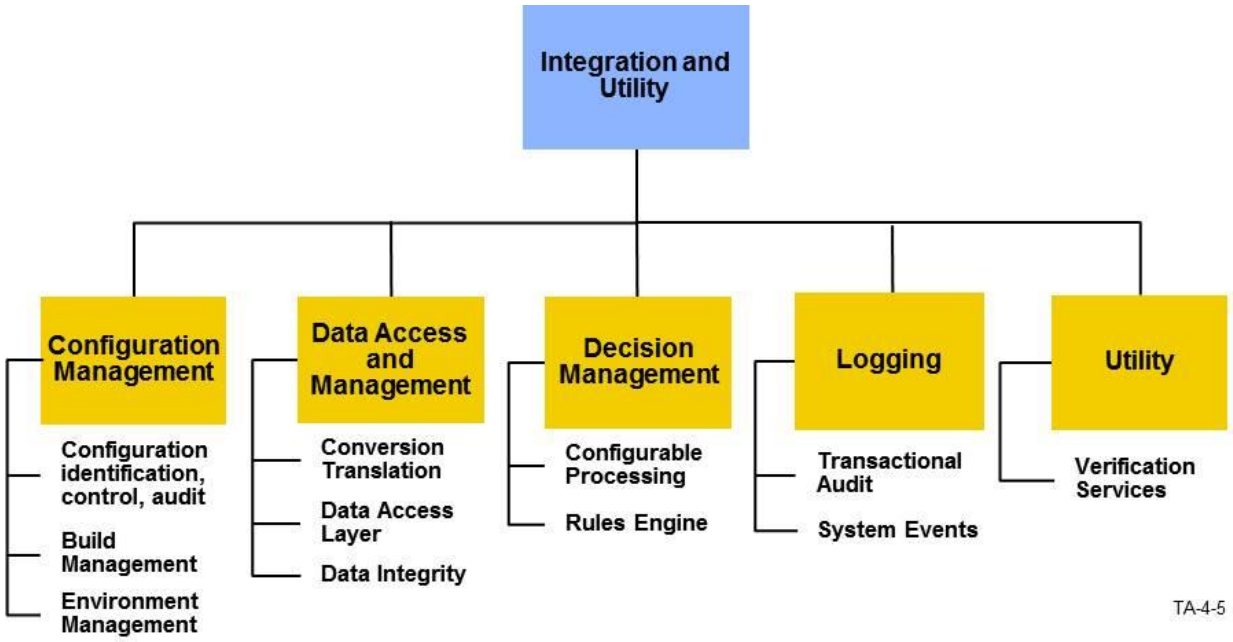

Figure 4-5. Integration and Utility TSCs

Technical Service Parts

This section defines the individual parts of a technical service. In the MITA Framework, a technical service has the same metadata as a business service. Both require the following associated data:

- **Service Name** – The name of the service component invoked by the Technical Service Definition Package (TSDP) that is a MITA-defined set of metadata describing the service. The service contract information describes the expected behavior of the interface (e.g., whether the interface is real-time or an online interface), as well as the security and privacy constraints on the service (i.e., requires strong encryption). The following are examples of interface behavior patterns:
  - A one-way interface only receives or outputs data (e.g., report generator).
  - A two-way interface receives and sends data. Two-way traffic has two other attributes: (1) initiator defines who initiates the interface, either the service (e.g., request for information) or the outside client (e.g., inquiry), and (2) processing characteristic defines the relationship between the input and the output:
    - **Point-of-sale (POS) transaction** – A real-time transaction (e.g., a pharmacy POS) that features very constrained response time and high reliability.
    - **Online transaction** – An inquiry on a provider that features more relaxed response time while still allowing for conversation-type human interaction.
    - **Batch** – Typical batch processing constraints.

- **Asynchronous** – No constraints on processing times but require response and data coordination.

- **Purpose** – The technical area's principles and TCM depict the purpose.
- **Business Logic** – Initially, the business logic is free-form text or template driven (e.g., MITA Business Process Template), application developers will eventually define and code business logic in order to incorporate into business process orchestration steps and standardized business rule definitions. States should ensure the use of business rules engines to separate business rules from core programming, and should provide information about the change control process that will manage development and implementation of business rules. States should be able to accommodate changes to business rules on a regularized schedule and on an emergency basis.
- **Constraints** – Lists constraints of the service, e.g., loose authentication control.
- **Formal Interface Definition** – Defines the services triggers and results. It also documents the interfaces and operations the service uses while using Web Service Definition Language (WSDL). Initially, the MITA Framework provides an informal textual description (i.e., template) of the services interface. As stakeholders develop the MITA services, they also develop the WSDL interface descriptions.

WSDL is a document written in Extensible Markup Language (XML) and is an XML document. The document describes a Web service. It specifies the location of the service and the operations (or methods) the service exposes.

- **Use Case** – The architect or designer writes the use case in free-form text and supports the text with Object Management Group (OMG) Unified Modeling Language (UML) models prior to the development of the business services.
- **Solution Set** – Maps the solution set(s) currently available to execute this service. Solution sets are previously defined groupings of technical services that are operational.
- **Structure and Activity Diagrams –** Graphically depicts the business logic performed by the service and interconnects the solution sets. The architect or designer models the diagram in OMG Business Process Model and Notation (BPMN). Using a modeling tool based on UML notation produces an UML diagram as part of the set.
- **Performance Standards** – Defines the anticipated performance standards of the service so all stakeholders measure the same metrics in the same way.
- **Test Scenarios and Test Cases** – Validates documentation of compliance to the service contract.
- **Map to MITA Data Models** – Maps a trace of data used by the service (e.g., data in motion and shared business data only) to the MITA Conceptual Data Model (CDM) and Logical Data Model (LDM). This is an enhancement of the mapping done for the business service. The XML schema defines the data in motion. American National Standards Institute (ANSI) Accredited Standards Committee (ASC) X12 positional

transactions and World Wide Web Consortium (W3C) XML Schemas are examples of defined data in motion.

Technical Service Development

The process for developing a technical service is the same process for developing a business service.

The first step in developing a technical service is to determine if the service already exists. Since there is not a service registry in place, the current process entails contacting the MITA review entities (e.g., National Medicaid EDI Healthcare (NMEH) workgroup) and CMS to request information pertaining to existing technical services. A CMS MITA Technical Service registry does not currently exist, but planning is underway. CMS will release more information it establishes registries for technical services. Further definition of technical service registries and how to provide the technical services to a national repository will follow in subsequent guidance.

- If the service does exist, the systems designer looks at the associated metadata to determine whether to adapt or extend the service.
- If the technical service does not exist in a usable fashion from any other state, then the stakeholder creates the technical service and TSDP. Most of the information already exists in the business service requirements, technical area principles, and technical TCM.

The process for creating a TSDP is similar to that of creating a Business Service Definition Package (BSDP) (see Part III, Chapter 3, Business Services). The differences between the two processes are below:

- **Service Name Development** – Use to register the technical service in the MITA service registry. The stakeholders determine the technical naming convention.
- **Service Contract Development** – Process involves the following elements:
  - The purpose is a short one- to two-sentence description of what the service does, and where it exists in the business service requirements, technical area principles, and TCM.
  - The stakeholder develops the formal interface definition, using the required Triggers, Results, and MITA data model, documenting the interfaces and operations used by the service. Initially, the MITA Framework provides an informal textual description (i.e., template) of the services interface. As the MITA Framework matures, stakeholders develop more WSDL interface descriptions.

WSDL describes a Web service in two fundamental stages: one abstract and one concrete. Within each stage, the description uses a number of constructs to promote reusability of the description and separate independent design concerns.

At an abstract level, WSDL describes a Web service in terms of the messages it sends and receives. Messages are independent of a specific wire format using a type system, typically XML Schema.

An operation associates a message exchange pattern with one or more messages. A message exchange pattern identifies the sequence and cardinality of messages sent to and/or received from. An interface groups together operations without any commitment to transport or wire format.

At a concrete level, a binding specifies transport and wire format details for one or more interfaces. An endpoint associates a network address with a binding. Finally, a service groups together endpoints that execute a common interface.

Figure 4-6 below is a sample WSDL.

```
<?xml version="1.0" encoding="UTF-8"?><wsdl:definitions name="MemberEligibility"
    targetNamespace="http://member.mitaexchange.org/Services/MemberEligibility/"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://member.mitaexchange.org/Services/MemberEligibility/"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsd_1="http://member.mitaexchange.org/Messages/">
  <wsdl:types>
    <xsd:schema targetNamespace="http://member.mitaexchange.org/Services/MemberEligibility/"
                xmlns:xsd_1="http://member.mitaexchange.org/Messages/">
      <xsd:import namespace="http://member.mitaexchange.org/Messages/" schemaLocation="../Messages/Messages.xsd"/>
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="InquireMemberEligibilityRequest">
      <wsdl:part element="xsd_1:InquireMemberEligibilityRequest" name="request"/>
  </wsdl:message>
  <wsdl:message name="InquireMemberEligibilityResponse">
      <wsdl:part element="xsd_1:InquireMemberEligibilityResponse" name="response"/>
  </wsdl:message>
  <wsdl:portType name="MemberEligibilityM3">
      <wsdl:operation name="InquireMemberEligibility">
          <wsdl:input message="tns:InquireMemberEligibilityRequest"/>
          <wsdl:output message="tns:InquireMemberEligibilityResponse"/>
      </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="MemberEligibilityM3Binding" type="tns:MemberEligibilityM3">
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
      <wsdl:operation name="InquireMemberEligibility">
        <soap:operation soapAction="http://member.mitaexchange.org/Services/MemberEligibility/InquireMemberEligibility/"/>
        <wsdl:input>
          <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <soap:body parts="response" use="literal"/>
        </wsdl:output>
      </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="MemberEligibilityServiceM3">
      <wsdl:port binding="tns:MemberEligibilityM3Binding" name="MemberEligibilityM3">
          <soap:address location="http://member.mitaexchange.org/Services/MemberEligibility/MemberEligibilityServiceM3"/>
      </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

XML Schema

Message

Operation

Transport

Binding

Common Interface

TA-4-11

Figure 4-6. Sample WSDL

A solution set is a combination of resources that documents or describes a specific implementation of a business or technical service. MITA solution sets map to both business and technical services to assist States with implementation planning.

Technical Service Solution Sets

Since technical services are implementation-neutral, the MITA Framework requires a method for documenting the execution details. This is so that States and vendors do not

have to keep recreating the solution for the service. Solution sets are the logical deployment of a MITA service. The solution sets are pattern-specific and can be platform- and technology-dependent:

- A solution set is a deployment of a technical service.
- Solution set mapping is in **Figure 4-7.**
- Since there is not a solution set registry in place, the current process entails contacting the MITA review entities (e.g., NMEH) and CMS to request information pertaining to existing technical services.
- States can use MITA Solution Sets to determine whether there is already a deployment of a MITA service applicable to their specific environment.

Figure 4-7. Conceptual Relationship – Technical Function to Solution Sets

A technical service solution set consists of the SMA implementation-specific TSDP. The SMA derives the TSDP from the MITA TSDP and adds the specifications to the MITA TSDP attributes (e.g., protocol,binding information, and endpoint). The state's TSDP provides the specifications for the technical service the SMA is deploying. In some cases, a code generator may use the WSDL in the state's TSDP to actually generate some of the required code. The system designer of the technical service (i.e., the state or vendor) is responsible for producing the TSDP for this solution if one does not already exist in the MITA repository.

Technical Services Flow

The technical service's objective is to provide an independent version of a technical capability that can merge with other services to form composite business processes. Services guided by the following architectural concepts provide this independence:

- Loosely coupled services.
- No predefined predecessor or successor services to an individual service.
- Services configured using a service contract and an orchestration language.
- Changes to the flow of services made through changes to this orchestration, not to the service itself.
- Mandatory interface compatibility between the services.

Figure 4-8 depicts an orchestrated flow of services using the conceptual service layout above. This diagram does not provide details regarding the feedback loop, but actually shows a simple combination of services within the service areas mentioned.

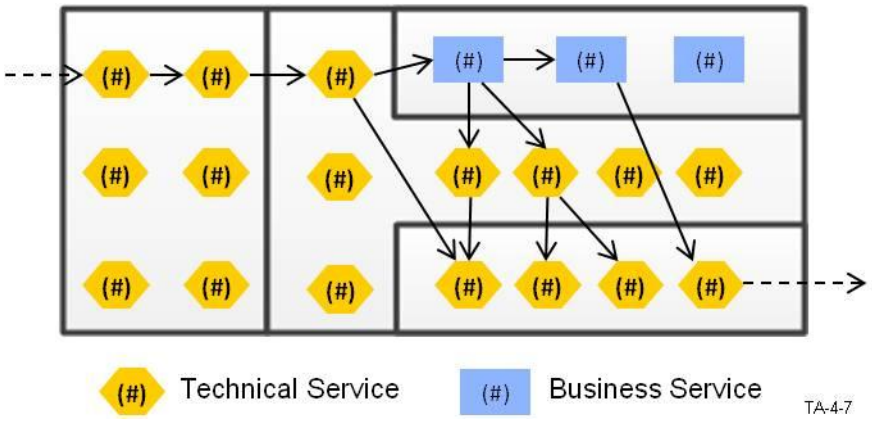

Figure 4-8. Service Orchestration Using the Conceptual Layout

Arguably the biggest benefit of an SOA is the ability to add and/or replace services with other services, provided it has the same signature (input and output), and without affecting any of the other connected services or requiring their modification.

Figure 4-9 below illustrates the addition of a technical service into a previously defined orchestrated process. The above-listed service characteristics allow this achievement.

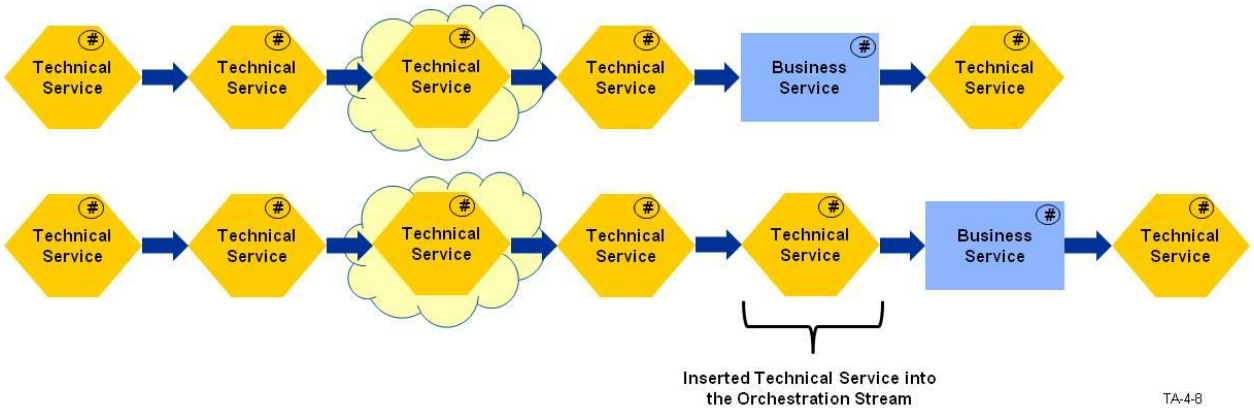

Figure 4-9. Process Orchestration Modifications

As stated earlier, the service contract defines access to an individual service using WSDL. The process to define a flow linking several services together, called orchestration, typically uses Business Process Execution Language (BPEL) or OMG BPMN.

BPEL or BPMN describe the behavior of a Business or technical service based on interactions between the service and other services. The interaction with each service occurs through the service interfaces, and service interface link encapsulates the structure of the relationship at the interface level. The BPEL/BPMN defines how multiple service interactions coordinate to achieve a business goal, as well as the state and the logic necessary for this coordination. Finally, BPEL/BPMN also introduces systematic mechanisms for dealing with exceptions and processing faults.

Service Orchestration defines successor and predecessor services to a service. Since this orchestration bases itself on definite deployment specifics, the MITA Framework does not develop the BPEL or BPMN orchestration for a service.

The SOA service composition (a.k.a. composite service) contains services assembled to provide the functionality required to automate a specific business task or process.

Figure 4-10 depicts a composite service that combines a business service with two technical services.

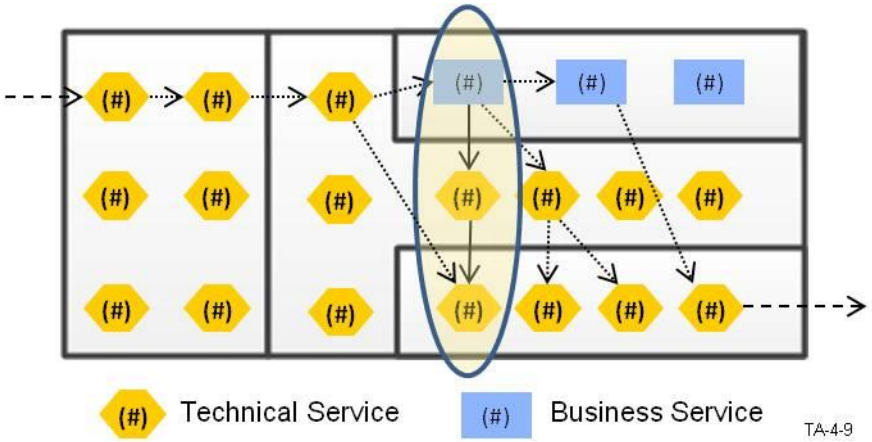

Figure 4-10. SOA Composite Service

The Federal Chief Information Officers Council *A Practical Guide to Federal Service Oriented Architecture* (Version 1.1 June 2008) contains additional relevant SOA-specific information. Many sections of the document are directly applicable to the MITA initiative. The following figure from the document is especially pertinent since it illustrates the accumulation of interdependent aspects necessary to achieve a target Service Oriented Enterprise, a primary MITA objective (See **Figure 4-11).**

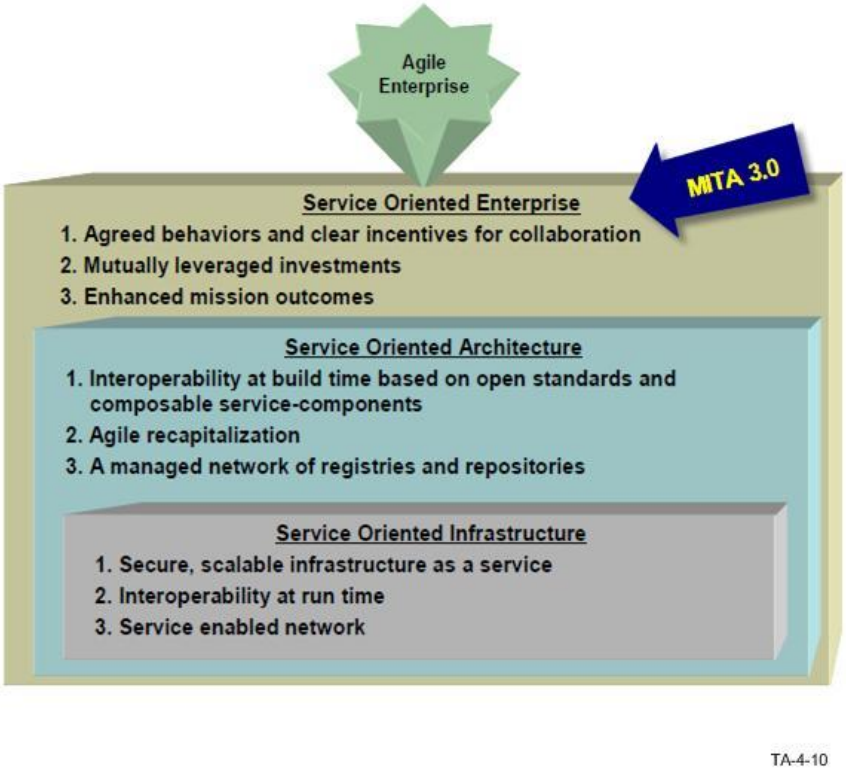

Figure 4-11. Federal SOA Framework

Benefits of Technical Services

The use of technical services provides the following benefits:

- **Reuse** – Multiple organizations and systems share the service. Each time an additional system uses the service, it adds to the value of the original investment in that service. An example is security access and controls, a service needed by every application. By writing this service once and reusing it for all applications, stakeholders reduce the development time and complexity.
- **Cost** – Service written once and maintained in one place reduces overall development and maintenance costs. An example is security access and controls, costs increase when each application writes its own security access and control modules.
- **Consistency** – When multiple applications share a service, the results are always the same. Using the security access and control example, users benefit from using the same password to access multiple systems. Another example would be performance standards services. By always measuring performance in the same way, it is possible to perform valid comparisons between organizations or within a single organization over a period.
- **Flexibility** – Widespread use of services makes it possible for systems to be more responsive to change. Changes occur by improving the functionality of a single service, using a different service to accomplish a task, or incorporating new services.

- **Broader Market Place** – The TA defines its technical services in general industry terms not specifically tailored to the Medicaid environment. This allows technical solutions from non-Medicaid environments in the MITA Medicaid Enterprise.

In addition, MITA Technical Services support the SMA alignment with the MITA enterprise architectures. The TA defines the MITA Technical Services in terms of common solutions that enable state-specific deployments, making it possible to develop adaptable and extensible services. The MITA Technical Service approach combined with the accommodation for state-specific deployments means that the MITA services help meet the SMA business needs in a rapidly changing Medicaid environment.

Using Technical Services

The MITA Technical Service is a reference document that identifies:

- The technical interface exposed to other processes.
- The standard interface definition.
- A description of the underlying business logic.