**PROJECT POPLIN**

**MEDICAID REFERENCE ARCHITECTURE**

**CARE/CASE SERVICE DEFINITIONS**

**MITA Governance Board**

**Poplin Working Group**

**State of Vermont**

**2018.02.23**

# POPLIN Project Background

The Centers for Medicare and Medicaid Services (CMS) seeks to develop a robust module marketplace for Medicaid to promote competition of solutions, reduce risk, streamline certification, and better serve the needs of individual states’ Medicaid Enterprise Systems (MES) through standards and conditions for its funding of state implementations. Such conditions include: application of modular design, development, and deployment methodologies and technologies for state-specific MES components and reuse of those components when possible.

To help states and vendors prepare for and use a module marketplace for Medicaid, CMS has sponsored efforts with MITRE Corporation through its Federally Funded Research and Development Center’s (FFRDC) CMS Alliance to Modernize Healthcare (CAMH) to develop a reference architecture (Project Poplin) that can extend the Medicaid Information Technology Architecture (MITA).

Poplin seeks to provide standard definitions for application services, enabling MES implementations across a diversity of state requirements so that vendors can develop robust technologies with broad application. This Service Definition Template defines the components required in a service definition document.

# Introduction

The purpose of this document is to decompose Care Management business processes, to identify the dependencies between all systems in a modern /modular Medicaid Enterprise.

The functions of Care Management can be accomplished through the implementation of a Case Management system. A Case Management system enables an organization to bring all services for a client into a common platform giving a complete view of a client and their services.

We will be focusing on Case Management. How it fits into the MITA architecture, the processes it enables, and the standard interfaces that need to be available for a more connected enterprise.

# Modularity

There are many levels of modularity that the MES’s could be abstracted to. Each module fulfills a business need of the overall MES. The 10 Business Areas as defined by MITA are:

* Business Relationship Management
* Care Management
* Contractor Management
* Eligibility and Enrollment Management
* Financial Management
* Member Management
* Operations Management
* Performance Management
* Plan Management
* Provider Management

Many business processes cross over the above abstractions. This can make it difficult when attributing a process to a predefined business area. Each of these areas also contains others. Case Management is contained in the Care Management business area.

# Assumptions

## Security

A Zero Trust security model will be applied to both internal and external service access.

## Systems

The Case Management system requires/includes:

1. Identity Management
2. Role Based Access Management
3. Consent Management  
   responsible for maintaining consent form entry and analysis, and the client’s data availability preferences.
4. Master Person Index (MPI)  
   MPI contains only the information needed to identify a person across all systems, enabling enterprise wide analysis. Every Client and Provider would have their basic Person information here. No other system stores information in the MPI.
5. Master Client Index (MCI)  
   A person listed in the master client index is receiving, or has received, services from the State. MCI maintains the client health information and information specific to other human service programs.
6. Document Management
7. Communications - Noticing
8. Communications - Messaging
9. Human Service Program Management (HSPM)  
   HSPM manages plan information , including: the services offered, the rates allowed, the benefits provided, and other information that governs the program is maintained here.
10. Provider Management  
    PM is responsible for the list of providers, what roles that can fill, and what services they can perform.

## API’s

API’s will be defined using Representational State Transfer (REST) design standards. The use of the PATCH operation will not be included in the definitions below. POST, GET, PUT, and DELETE operations will be defined as appropriate.

The RAML language will be used to detail these REST API’s.

## Data Standards

Where applicable we will use the data-standards outlined in the standards referenced below. Preference should be given in the order listed.

1. [Standard Health Record](http://standardhealthrecord.org/shr/)
2. [Fast Healthcare Interoperability Resources](http://hl7.org/fhir/)
3. [Health Level Seven](http://www.hl7.org/Special/committees/index.cfm?ref=common)
4. [National Information Exchange Model](https://www.niem.gov/)

# Communications

While identifying the services needed we also need to define their method of communication.

There are three ways systems can communicate with each other

1. Batch Interface  
   This is a scheduled process of transferring a set of data between systems. Best suited to exchange reference data that does not change frequently. One example would be a list of ICD-10 codes.
2. Real-Time Interface  
   This type of interface is necessary when requesting time sensitive information or the most accurate data in an ever-changing environment. One example of this would be a request to see if a client is currently enrolled, or eligible, to receive a service.
3. Event Driven Interface  
   This is a key component of any enterprise. When data is changing in multiple systems there is a need for notifying others of changes that could affect their activities. One example would be if a provider was no longer eligible to perform services we would want the Case system to be aware of this as soon as possible. In this type of environment, a message would be generated, and the Case solution would see the message and process it as needed.

Solutions expose functionality in multiple ways to handle these types of communications. Publishing Application Programming Interfaces (APIs), Service Oriented Architecture (SOA) Services, and a Messaging Service. These methods allow multiple systems to communicate in a standard way and also enables the realization of a more modular enterprise.

# System Approach

This project has opted (assuming writeup above states it) to utilized an integration layer to enable real-time and event driven communication between systems. The t key components of integration layer are an API Gateway and a Messaging system.

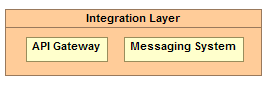


Figure 1 - Integration Layer

The API Gateway is used to expose the APIs needed to support real-time access to information and processes. The API’s define: how systems will communicate, the format of the request, what the expected response will be, and the protocol used to access the API.

The Messaging System allows for the real-time processing of information which (?) enables event processing.

Through the integration layer, a system could take action based on an event, such as the approval of an application and trigger additional processes.

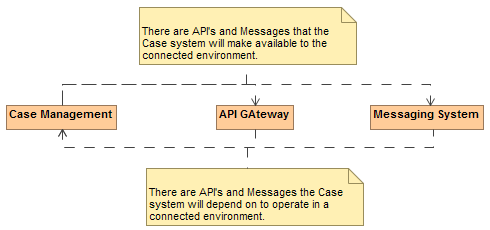


Figure 2 - API and Message Connectivity

Although we will mention the APIs and Messaging which relate to the system , we will only be defining the details for those that the case system provides. Any system that adheres to the following specifications will be able to communicate with the case system using the APIs and Standard Messaging.

# System Dependencies

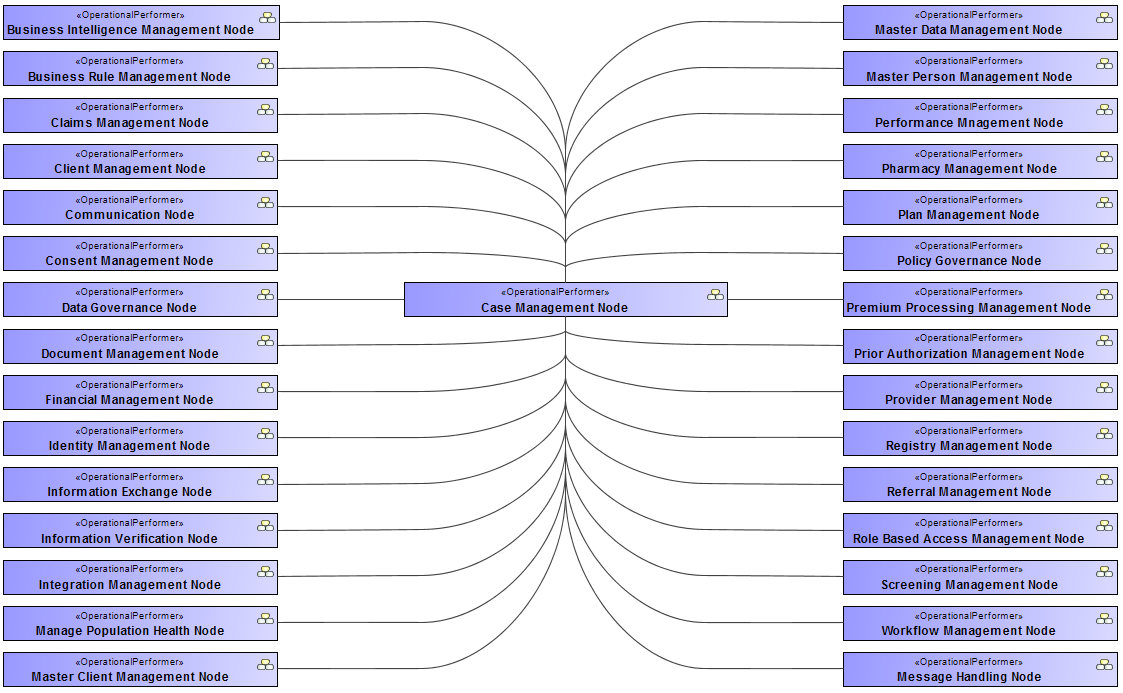


Figure 3 – High Level Care and Case Dependencies

As seen in Figure 3, Case Management. It is essential that the proper sub-systems are in place to make the migration to a common modular future state possible, progress in a logical fashion, and reusable.

# Case Activities

A generally accepted list of high-level case functionality includes:

| **High Level Case Activities Table** |
| --- |

| **Name** | **Documentation** |
| --- | --- |
| Initiate | processing incoming referrals and other information that are the basis for case creation. |
| Engage | Engagement establishes the relationship and sets the ground rules which are enhanced and reviewed throughout the phases of case management. |
| Assess | Assessment is a dynamic and ongoing process where information is gathered from a range of sources about a person, including their life situation, and formal and informal supports. The range of information is then considered in the context of information and advice from the person with a disability, their family and/or careers and informal supports |
| Plan | A plan is a map of actions that documents the issues, methods, responsibilities and timeframes needed to meet the identified goals. |
| Implement | Implementation is the process of putting into action the plan developed by the case manager, together with the person, their family and other supporting team members. |
| Monitor | Monitoring is an active and ongoing process where aspects of the planning and implementation phases are reviewed. It identifies the effectiveness and relevance of planned goals, focusing on the timeliness and success of strategies being used to achieve these goals. Additionally, it provides the opportunity to adjust the plan to address any unanticipated problems. |
| Review | Case management practice involves regular formal and informal review processes to ensure outcomes are relevant to needs. |
| Transition | Transition involves creating a plan with the clients and other supports to ensure they receive the care needed. |
| Close | Cessation of involvement by a case manager or the closure of a case may be influenced by many factors. These factors may relate to organizational or agency requirements in which a limited time frame is identified for the case manager involvement with any one client. This may raise significant concerns or questions as to the appropriateness of terminating contact. The decision as to whether to cease involvement or close the relationship can be influenced by a number of factors. |

| **Supporting High Level Case Activities Table** |
| --- |

| **Name** | **Documentation** |
| --- | --- |
| Analyze | Analyze data to determine where evidence shows changes are needed to improve outcomes, to detect fraud, and to verify utilization of services. |
|  |  |

The sections below will explore each of these areas to identify the following:

1. Identify the process(s)
2. Identify Dependencies on other systems / modules
3. Identify the Activities involved at the dependency boundary
4. Identify the objects necessary to support these activities
5. Identify the APIs the case module depends on being available
6. Identify the APIs the case module will make available
7. Identify Messaging the case module can act on
8. Identify Messaging the case will generate
9. Define the data objects needed to satisfy the API and Messaging defined
10. Define the resources needed to implement the activities, APIs, and messaging

The next two figures we have traced the high-level case processes to the applicable MITA Business Processes.

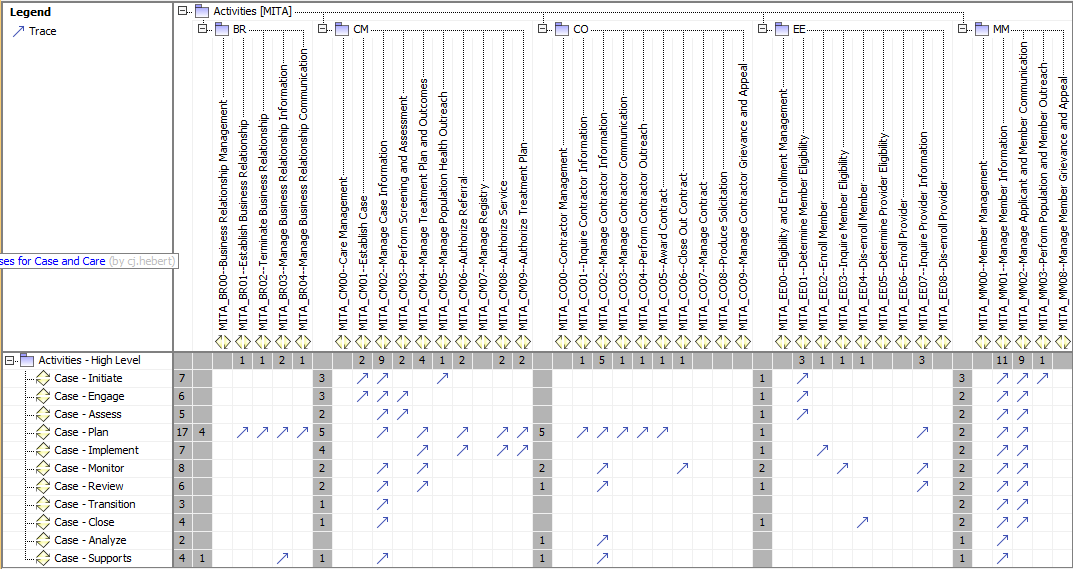


Figure 4 - Case to MITA Business Processes A

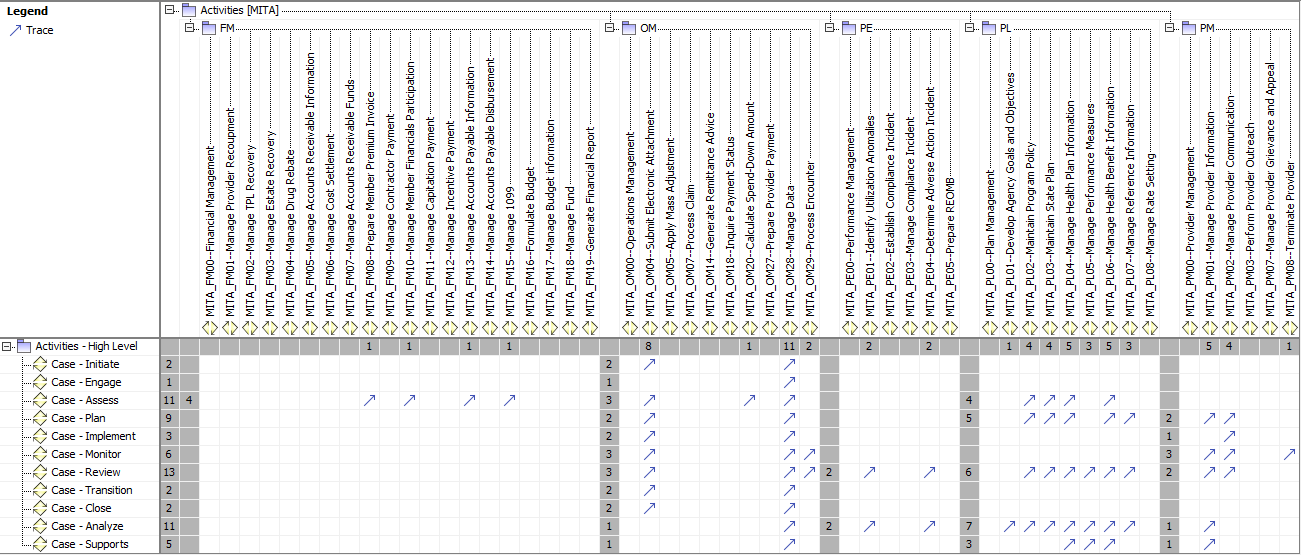


Figure 5 - Case to MITA Business Processes B

# Objects

# Code Values

## Case Type and Sub-Types

|  |  |  |
| --- | --- | --- |
| Case Type | Case Sub Type | Description |
| Human Services | Coordinated Care | Used to manage coordinated case services to a client |
| Human Services | TANIF | This case is for tracking the client’s activity in the TANIF programs |
| Human Services | LIHEAP | This case is for tracking the client’s activity in the LIHEAP programs |
| Human Services | Client Service Application | This case takes the submission of a client applications and tracks the processing and status. It may also trigger the creation of other cases as a result of the verification of the application information |
| Human Services | Provider Service Application | This case takes the submission of a provider’s applications and tracks the processing and status. It may also trigger the creation of other cases as a result of the verification of the provider application information |
| Human Services | Grievance |  |
| Human Services | Appeal |  |

## Case Detail Type and Sub-Types

|  |  |  |
| --- | --- | --- |
| Case Sub-Type | Detail Type | Description |
| TANIF | Service Eligibility Date | Used to manage coordinated case services to a client |
| TANIF | Service Acceptance Date |  |

## Case Element Type and Sub-Types

| Case Type | Case Sub Type | Description |
| --- | --- | --- |
| Human Services | Eligibility dates | The date range the client is eligible for a service |
|  |  |  |

# Services

The common services being defined below are to be realized as RESTful API’s.

A RESTful API is an application program interface (API) that uses HTTP requests to GET, PUT, POST and DELETE data.

A RESTful API -- also referred to as a RESTful web service -- is based on representational state transfer (REST) technology, an architectural style and approach to communications often used in web services development.

The API’s specifies how software components should interact. Additionally, APIs are used when programming graphical user interface (GUI) components.

In this section we will define the common API needed to support Case Management operations as well as the ones case system will supply for others to use.

## Security Services

.

As previously mentioned the necessary

These resources are shown in the resource diagram below:

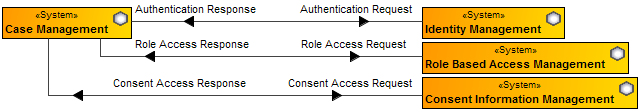


Figure 6 - Security Resources

All systems will be required to access the Identity Management resource to verify the identity of the user or system accessing information.

All systems will be required to interface with the Role Based Access Management to identify the processes the user may access.

And finally, when dealing with privacy issues all systems will be required to interface with the Consent Management system to verify the user has the permission of the client to access their data.

The following security services are necessary to enable controlled access to systems and data.

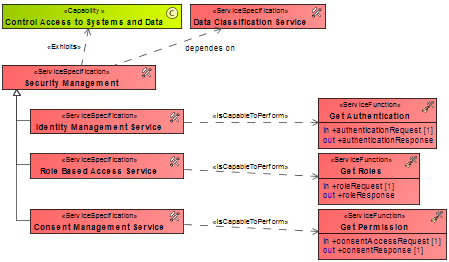


Figure 7 - Security Services

\* For additional details please see the specific service specification documents:

1. Identity Management Service Definitions
2. Role Based Access Control Service Definitions
3. Consent Management Service Definitions

## Person and Client Information Services

Case will not be directly responsible for maintaining Person and Client data. Case will be responsible for connecting to these resources to keep information in sync and use the Client Information Management system to store relevant case information.

The Case system will store and hold high-level case information necessary for Case Management functionality. Since the details of any given case may have information subject to various privacy standards it will be stored and managed in the Client Information Management system.

These resources are shown in the resource diagram below:

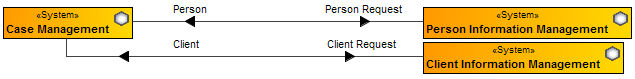


Figure 8 – Person and Client Resources

All systems will be required to link to the person and/or client entity to enable future analytics that can truly cross many barriers that exist today.

The following services are necessary at a minimum to enable consistent access to systems and data.

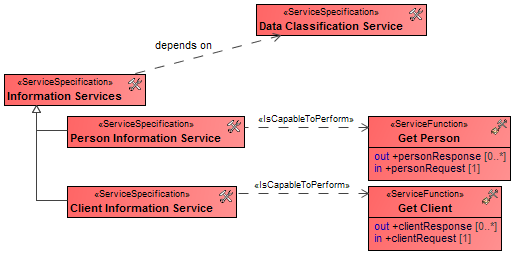


Figure 9 – Person and Client Services

For additional details please see the specific service specification documents:

1. Person Index Management Service Definitions
2. Client Information Management Service Definitions

In both cases the Case System will expect that if a person or client doesn’t exist then they should be created to allow the case system to establish the case.

## Document Services

The Case Management system will store, retrieve, and search documentation.



Figure 10 - Document Resources

Documents related to a case need to maintain that association. A method for searching for documents associated to the client should be searchable based on type and content.

Data classifications should be maintained to enable the proper level of security when accessing a client’s documentation.

It is expected that the document management system implements a standard document interface which has not been defined at this time.

Many document systems include proprietary API’s making necessary to have a specific set to be used by the systems in the MES.

Basic services include:

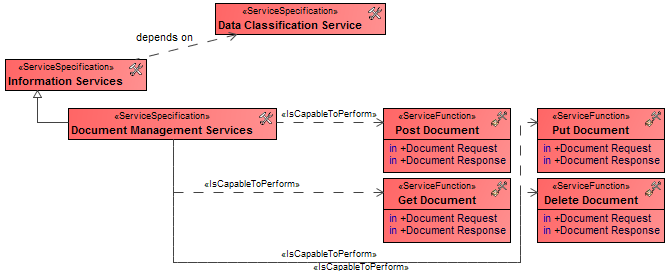


Figure 11 - Document Services

Please see the Document Management Service Definition documentation for more information.

## Communications

A Notice Management system is responsible for delivery, receipt, and cataloging of messages and their relationship to an entity such as a person, client, provider, case, or any other system involved.

The Message Management system is the component used to enable system to system communications in real time.

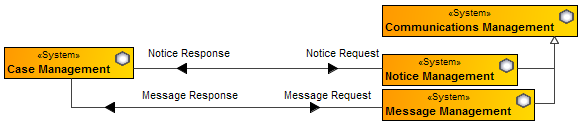


Figure 12 - Communication Resources

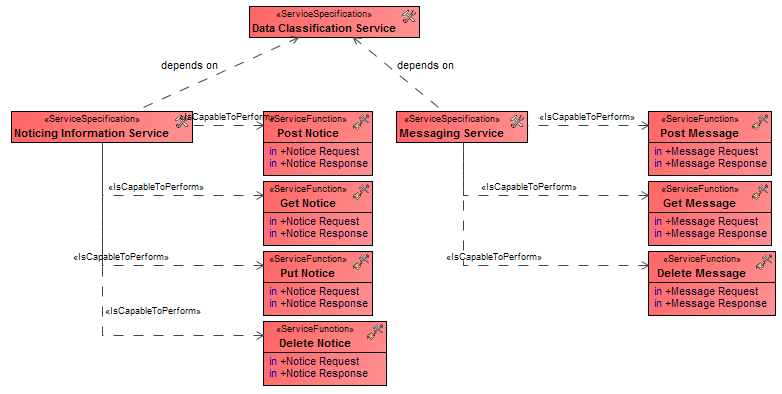


Figure 13 - Communication Services

See the Notice Management Service Definition or the Message Management Service Definition documentation for more information.

## Information Verification Services



Figure 14 - Verification Resources

\*The catalog of verifiable elements could be quite long and therefore we have only outlined those immediately needed by the case system.

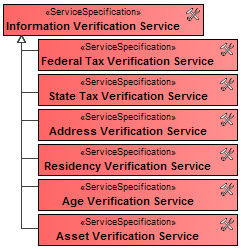


Figure 15 - Verification Services

See the Verification Service Definition documentation for more information.

## Provider Services

The provider services show provider availabilityas a primary care physician, what providers can provider what services, and to verify they have not been suspended or terminated from provided services.



Figure 16 - Provider Resources

The basic services required for case operations are shown below:

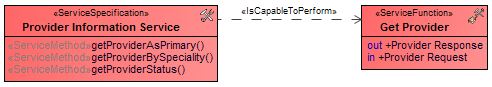


Figure 17 – Provider Services

See the Provider Service Definition documentation for additional details.

## Case Services

The services shown below are

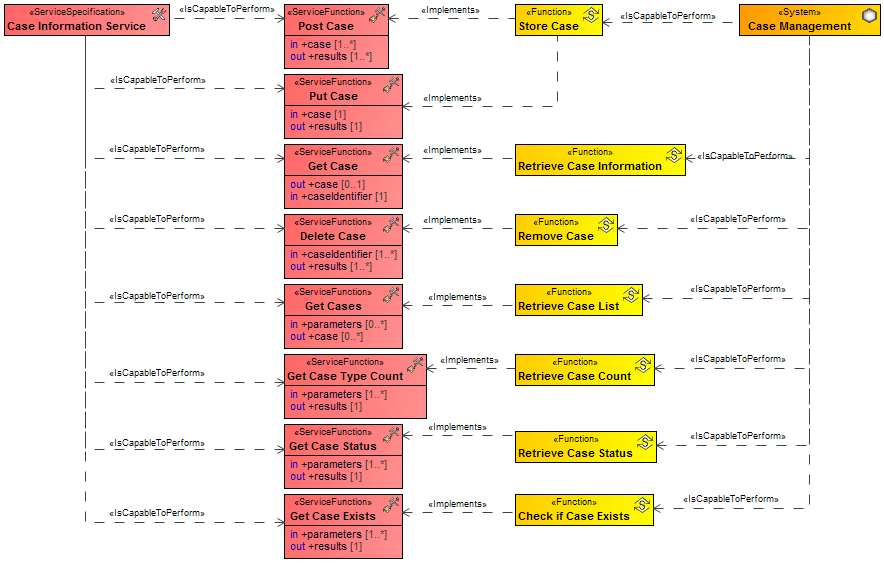


Figure 18 - Case Services

### Post Case

Case will be responsible for exposing a standard API to allow other systems to create a case.

For the purposes of expediting processes, a request to create a case by access the Post Case API should create a case if there is enough information available to know who the person involved is and what the issue is the case is needed to handle.

The process of case initiation will rely on Get Person and Get Client at a minimum.

As shown below, Post Case is being used by the Client Application Management.

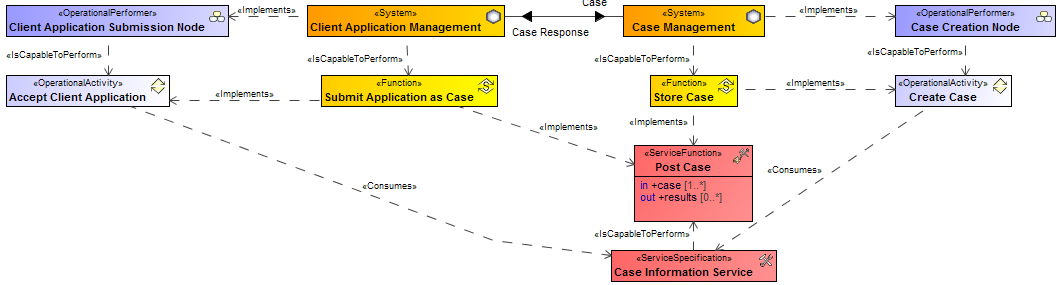


Figure 19 - Post Case

Many systems may access the Post Case API and some of the expected ones are shown below:

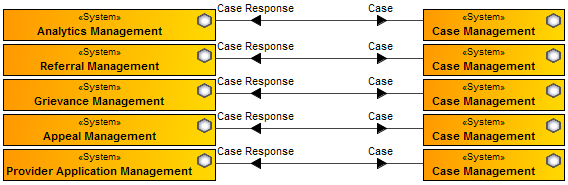


Figure 20 - Other Systems Accessing Case Creation

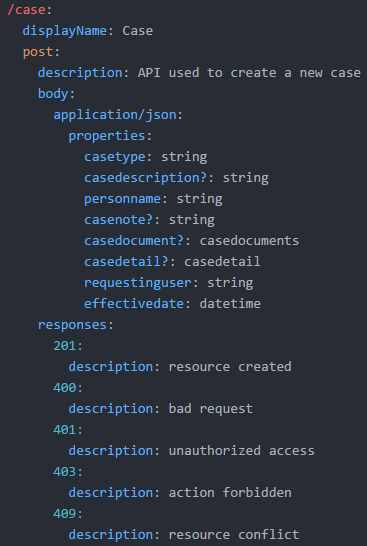


Figure 21 - Case Post API

### Get Case Exists

Case will be responsible for exposing a standard API to allow other systems to verify if a case of a specified type exists for a known person.

A link to the specific case will be returned for a given person and case type. The return will show if there is a past case or a active case.

As shown below, Get Case Exists is being used by Client Application Management to check if a case exists for a client application:

### Get Case Status

Case will be responsible for exposing a standard API to allow other systems to verify the status of a case.

The status may be returned given a person and a case type, or by a case identifier.

As shown below, Get Case Status is being used by Client Application Management to check the status of an application:

## Other Supporting Services Required

The services listed in this section are services that the Case System will be dependent on. It may be possible that the case system would implement some of these functions itself but in a fully modular approach these services would be provided by other systems. The services list below are mentioned here for the purpose of identifying the necessary services to support a modular case system.

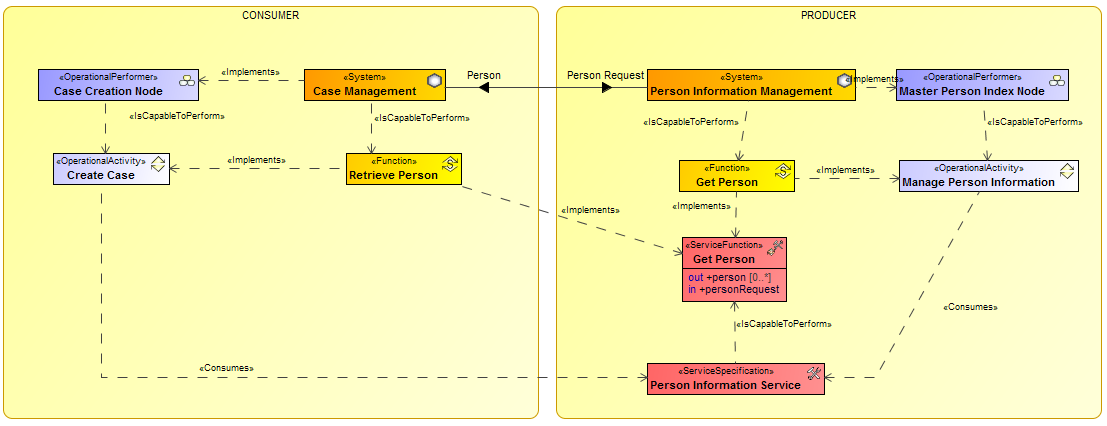
### Get Person

Case will rely on a Master Person Index. This module will be responsible for making available the service to get the basic information for a person.

This service will be defined in further detail in the Master Person Service Definitions document.

For case purposes, we should be able to get a person by a unique identifier, by information supplied such as name, address, phone, or by various other identification types.

Case would also expect if a person did not exist it should be created by the Master Person Index and flagged for review, so the case processes can carry on without interruption.



The service to Get Person should return:

1. A single person as requested
2. A list of possible matching persons as requested

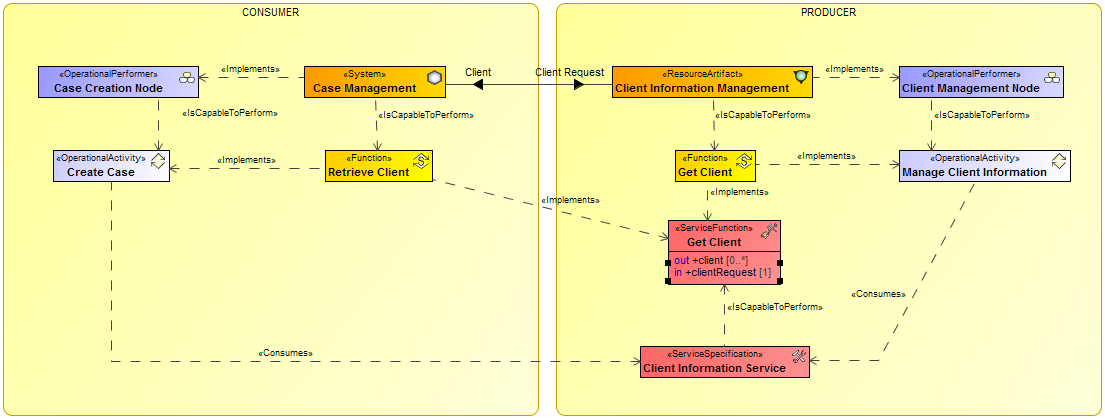
### Get Client

Case will rely on a Master Client Index. This module will be responsible for making available the services to get the basic information for a client.

This service will be defined in further detail in the Master Client Service Definitions document.

For case purposes, we should be able to get a client by a unique identifier, by information supplied such as name, address, phone, or by various other identification types.

Case would also expect if a client did not exist it should be created by the Master Client Index and flagged for review, so the case processes can carry on without interruption.



The service to Get Client should return:

1. A single client as requested
2. A list of possible matching clients as requested

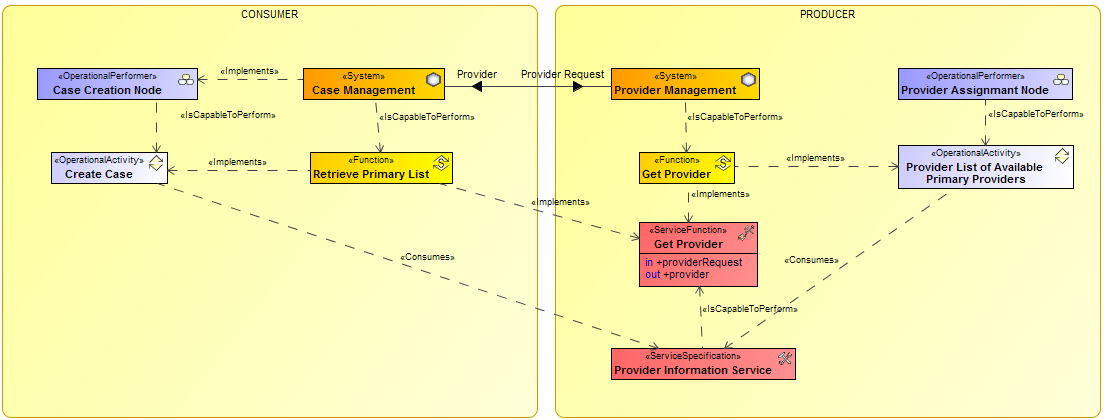
### Get Provider

Case will rely on a Master Provider Index. This module will be responsible for making available the services to get the basic information for a provider.

This service will be defined in further detail in the Master Provider Service Definition document.

For case purposes, we should be able to get a provider by a unique identifier, by information supplied such as name, address, phone, by role, by specialty, by geographical area, or by various other identification types.

There are no expectations on the Get service other than the stated above.



The service to Get Provider should return:

1. A single provider as requested
2. A list of possible matching providers as requested

### POST, GET, PUT, DELETE Plan

### POST, GET, PUT, DELETE Household

# Messaging

## Person Messages

Messaging would be expected from a Person system. Specifically, if a person was merged with another person record, the person address change, or some other change that could affect the possible delivery of services to a person.

## Client Messages

Messaging would be expected from the Client system. Specifically, if a client moved or some action was taken to indicate they may no longer qualify for services.

## Provider Messages

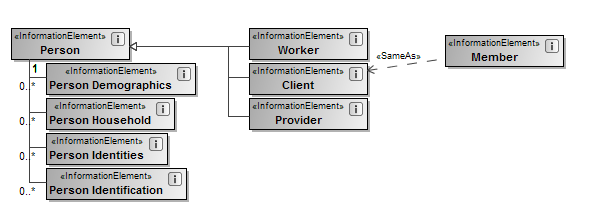
Messaging would be expected from a Provider system. Specifically, if a provider was terminated, specialty removed, or provider facility moved or was closed.

# Additional Object Information

## Person

The Person object is key for all systems. It holds the basic information of a “Person” and every other representation of a person (such as a client) starts with the basic person element.

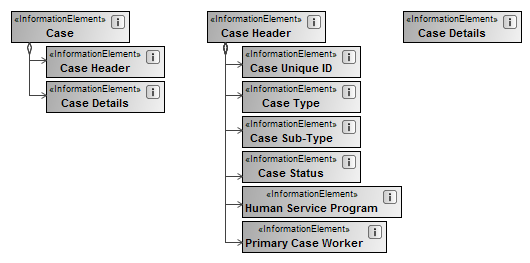
As shown below worker, client, provider are specializations of person. That is, they are a “person” with some additional information.



## Client

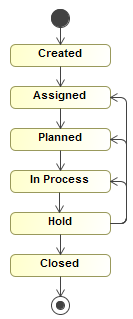
## Case

The basic structure



## Object States

## Case States

A case goes through many states during its life as shown here:  


# Appendix A - MITA Business Process Reference

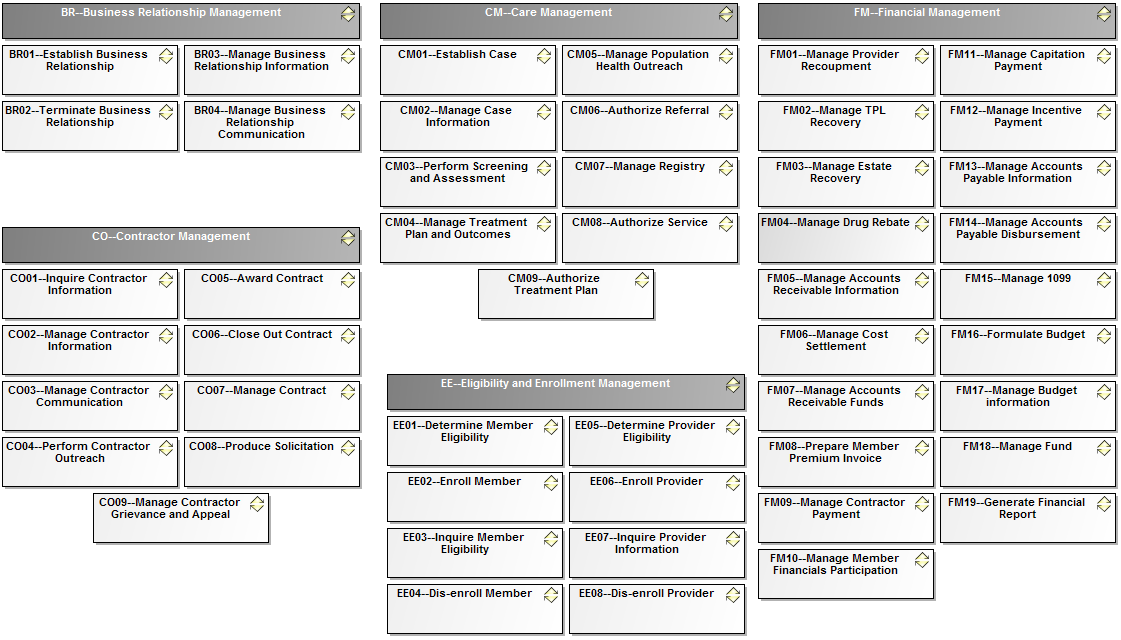


Figure 22 - MITA Business Processes A

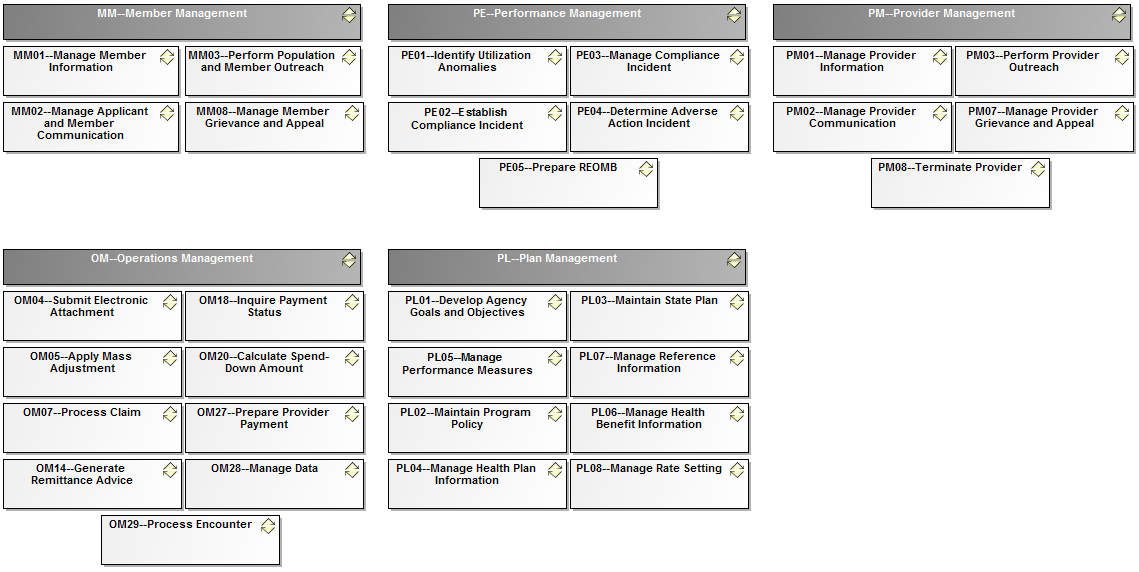


Figure 23 - MITA Business Processes B

# Appendix B - MITA Care Management Business Process Dependencies

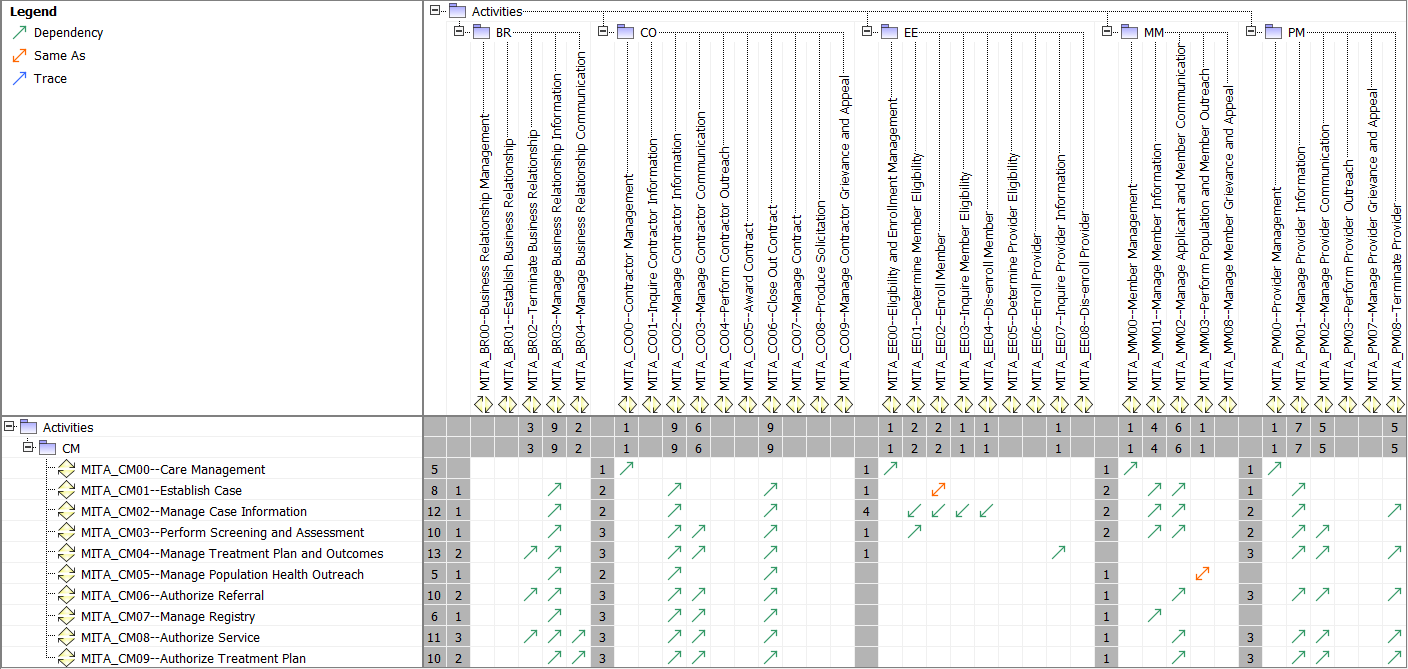


Figure 24 - MITA CM Dependencies A

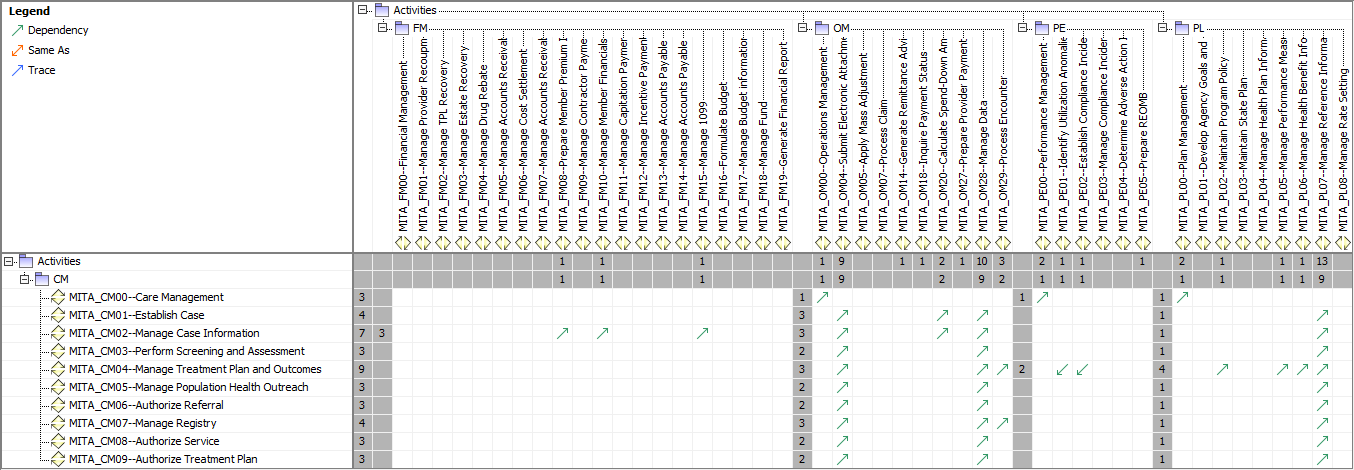


Figure 25 - MITA CM Dependencies B