

Service Specification Example: Nouveau Health Care Payment Manager Service

This document presents a sample specification for Nouveau Health Care's Payment Manager. For brevity, some of the sections are intentionally left incomplete. This material is taken from the book *Architecting Composite Applications and Services with TIBCO*.



<http://www.tibco.com>

Global Headquarters
3303 Hillview Avenue
Palo Alto, CA 94304
Tel: +1 650-846-1000
Toll Free: 1 800-420-8450
Fax: +1 650-846-1005

© 2012, TIBCO Software Inc. All rights reserved. TIBCO, the TIBCO logo, The Power of Now, and TIBCO Software are trademarks or registered trademarks of TIBCO Software Inc. in the United States and/or other countries. All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

Table of Contents

1 Service Overview	3
1.1 Service One-Line Description	3
1.2 Service Abstract	3
2 Service Context	3
3 Intended Utilization Scenarios	3
4 Interface Definitions	5
5 Referenced Components	6
6 Observable State	8
7 Triggered Behaviors	9
7.1 Deferred Payment Behavior	9
7.2 Settle Deferred Payments Behavior	10
7.3 Immediate Payment Behavior	11
8 Coordination	13
9 Constraints	13
10 Non-Functional Behavior	13
10.1 Performance	13
10.2 Availability within a Data Center	14
10.3 Site Disaster Recovery	14
10.4 Security	14
11 Deployment	14
Appendix A: Service Interface Specifications	14
Appendix B: Referenced Interface Specifications	14

1 Service Overview

1.1 Service One-Line Description

The Payment Manager manages both immediate and deferred claim payments to health care service providers.

1.2 Service Abstract

The Payment Manager manages both immediate and deferred claim payments to health care service providers. Under normal circumstances, requested payments are accumulated and paid periodically. The Payment Manager returns an immediate acknowledgement of the payment request, and sends an asynchronous notification of payment when the service provider account is settled. The scheduling of claim settlements is configurable on a per-provider basis. Under exceptional conditions, payment to the service provider can be made immediately and the returned acknowledgement indicates payment has been made. As this option places significant additional burden on the service, it should be used sparingly. The same interface is used to initiate both forms of payment.

2 Service Context

The usage context for the Payment Manager is shown in Figure 2-1. The service is intended to be used by the Claim Processor, and its operation is dependent upon the Provider Service, Banking Service, and Benefits Service. It also provides inputs to the Claim Tracker to maintain the overall status of claim processing, and thus is dependent upon it as well.

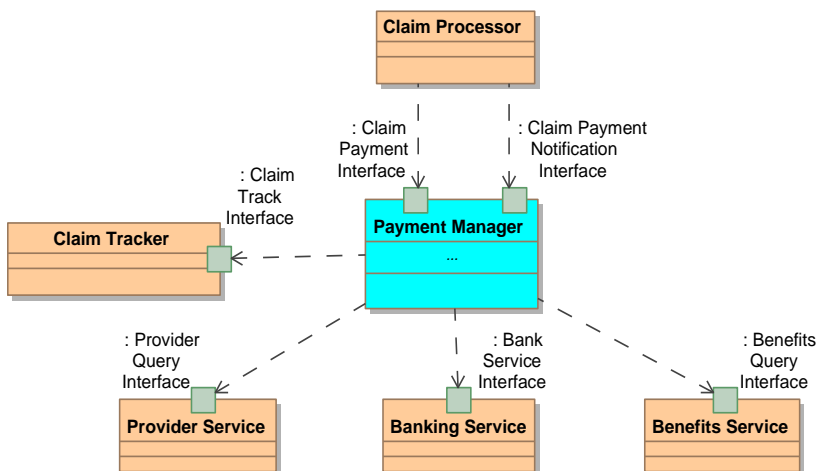


Figure 2-1
Payment Manager Usage Context

There can be more than one Claim Processor utilizing the Payment Manager. Nouveau Health Care has its own claim processor, but partner company claim processors may also utilize the Payment Manager.

3 Intended Utilization Scenarios

There are two intended utilization scenarios for the Payment Manager. The normal scenario is the deferred payment scenario shown in Figure 3-1. In this scenario, the service consumer directs the Payment Manager pay a claim and receives an acknowledgement that the request has been received. At some later point in

time (configurable for each health care service provider), the pending payments for that provider are paid and the payments are reported back to the service consumer. The scenario follows the delegation with asynchronous confirmation coordination pattern.

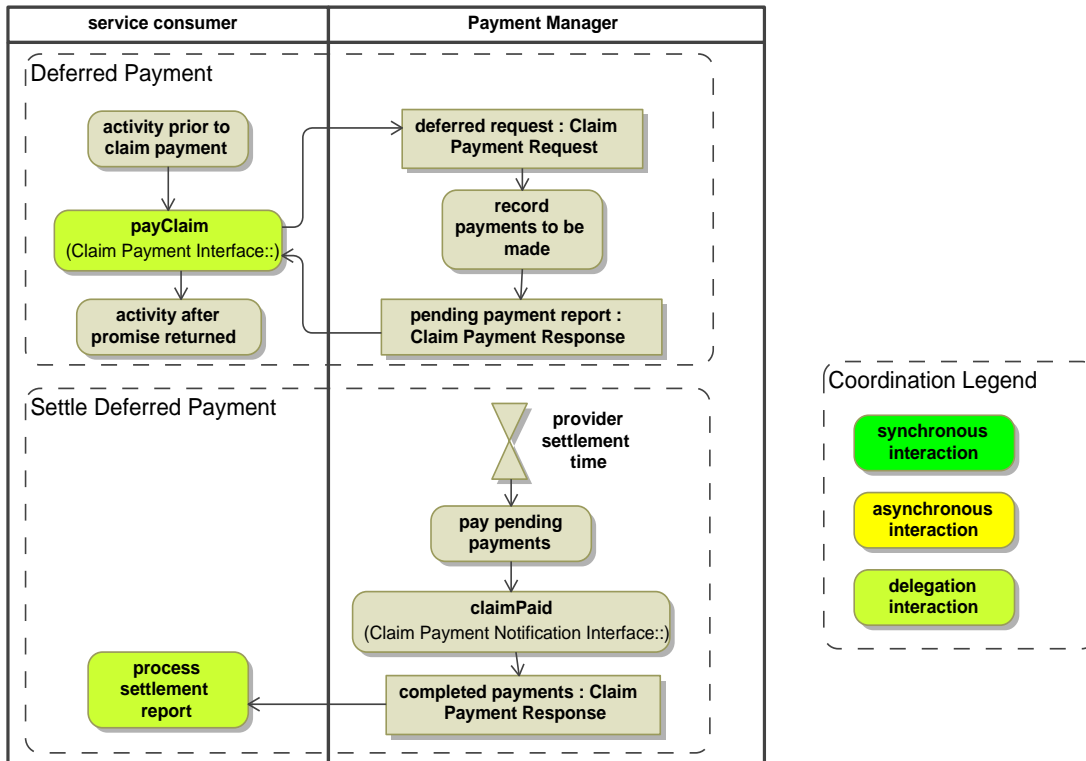


Figure 3-1
Deferred Payment Utilization Scenario

The second utilization scenario is the immediate payment scenario shown in Figure 3-2. This scenario is expected to represent a small volume of the overall service activity – less than one percent. In this scenario, the service consumer requests immediate payment and receives an acknowledgement that the specified claim payment has been made. This scenario uses a synchronous request-reply coordination pattern.

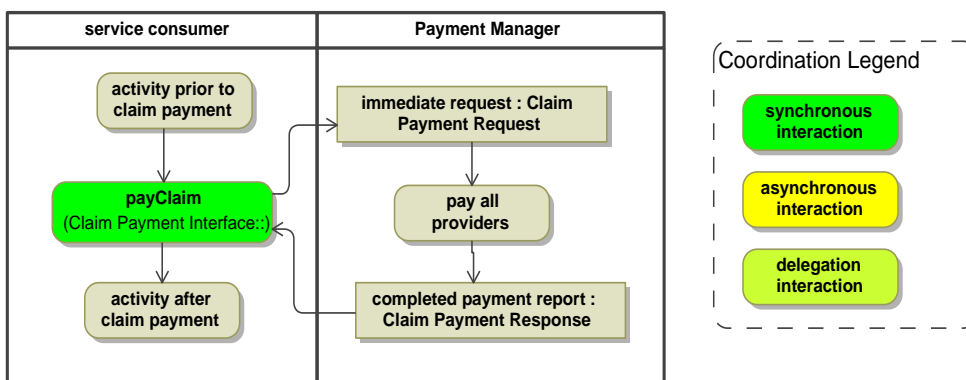


Figure 3-2
Immediate Payment Utilization Scenario

Both of these scenarios use the same payClaim() interface operation. The difference in behavior is determined by the value of the immediatePayment flag in the Claim Payment Request (Figure 4-1).

4 Interface Definitions

The Payment Manager presents two interfaces for use by service clients (Figure 4-1). The Claim Payment Interface provides the payClaim() operation, which is used by claim processors (and other relevant parties) to initiate claim payment. Depending upon the value of the Claim Payment Request's immediatePayment() flag, payment will either be deferred or immediate. For deferred payment, the Claim Payment Response indicates the pending amount for each service instance. For immediate payment, it indicates the amount actually paid. Access to the Claim Payment Interface will be via SOAP over HTTP (for partners) and JMS (for Nouveau Health Care).

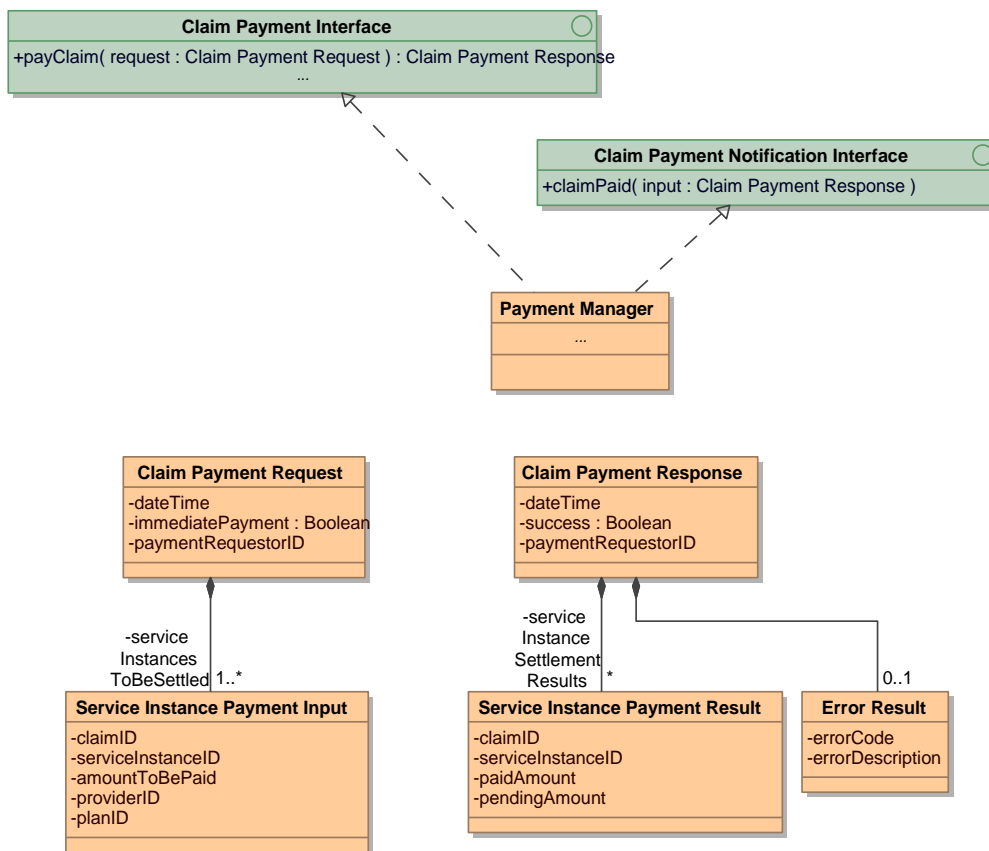


Figure 4-1
Payment Manager Service Interfaces for Claim Payment

The Claim Payment Notification Interface is used by the Payment Manager to notify service clients when deferred payments are actually made to the health care service providers. Such notifications can also occur as part of the recovery process after the failure of an immediate payment request. It is an XML over JMS interface. Subscription to this notification is accomplished at deployment time via JMS subscriptions. The details of this configuration are given in the Deployment Specifics section.

The other operations of the Claim Payment Interface are shown in Figure 4-2. *In the full specification the data structures associated with these operations would be documented here.*



Figure 4-2
Claim Payment Interface: Other Operations

5 Referenced Components

Figure 5-1 (taken from the Benefits Service Specification) shows the Benefits Query Interface `getPayerAccount()` interface used in this process.

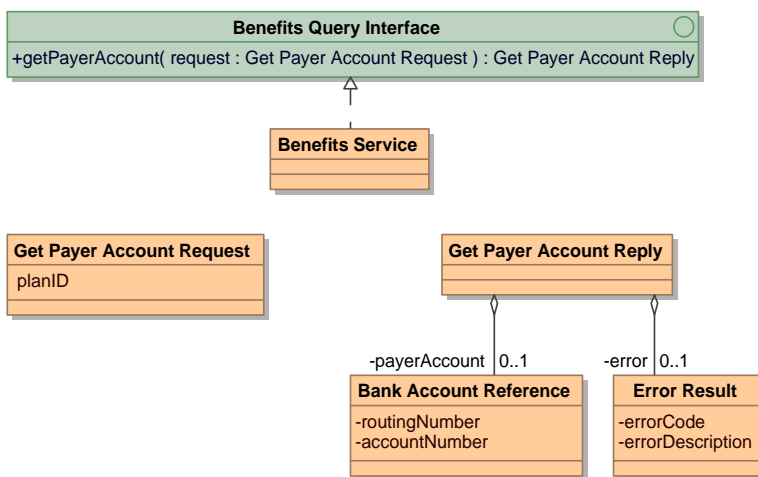


Figure 5-1
Benefits Query Interface - `getPayerAccount()` Operation

Figure 5-2 shows the Bank Service Interface (from the Bank Service Specification) that is used in this scenario and in the ones that follow.

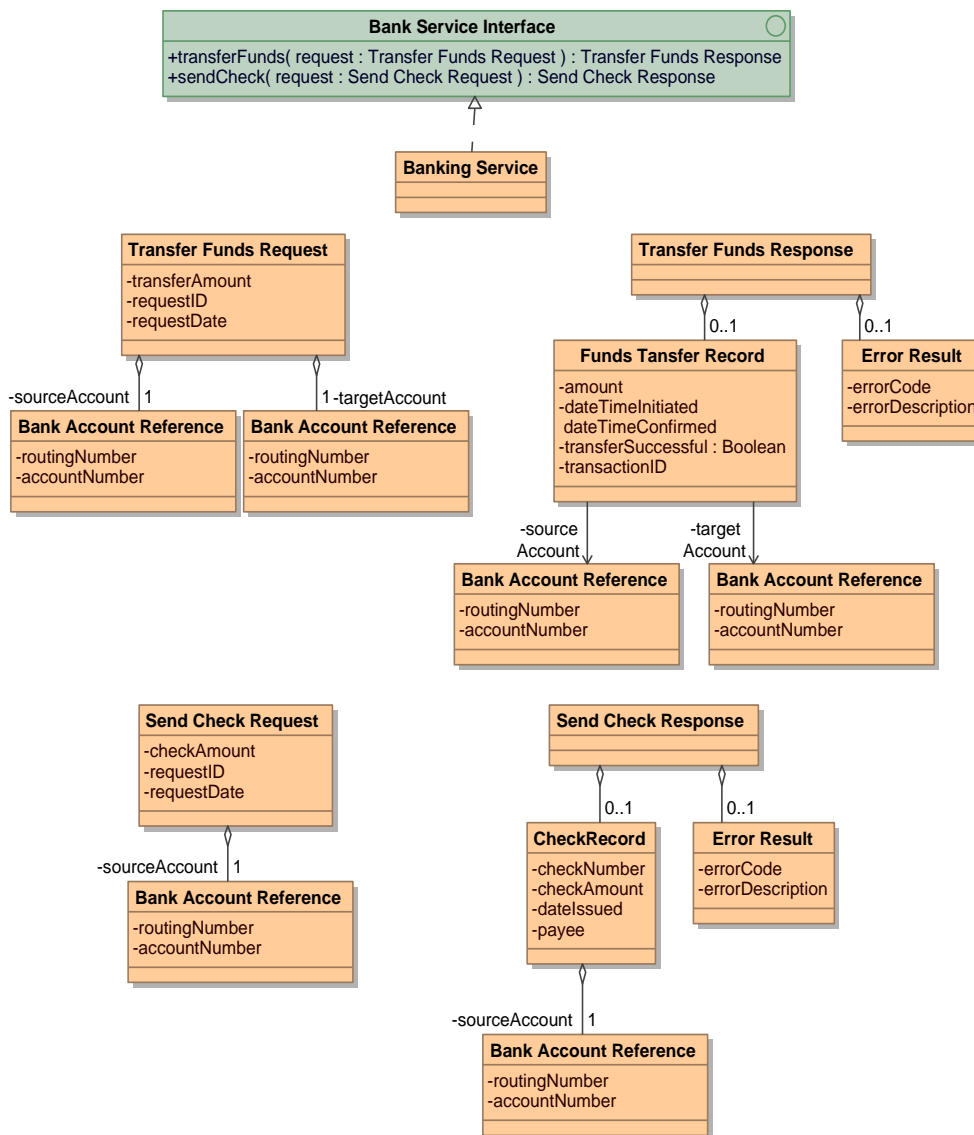


Figure 5-2
Bank Service Interface

Figure 5-3 shows the Claim Track Interface reportClaimStatus() operation used in this scenario and in others (taken from the Claim Tracker Specification).

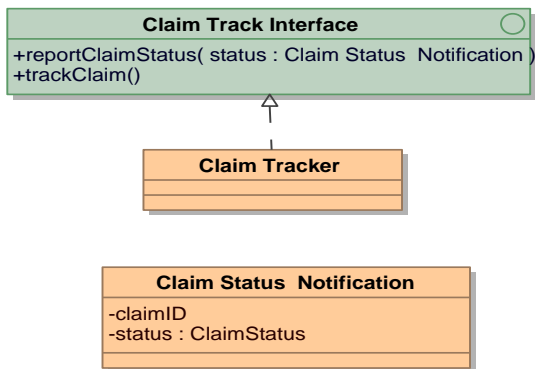


Figure 5-3
Claim Track Interface – reportClaimStatus() Operation

The interface for retrieving the provider account is shown in Figure 5-4. This is taken from the Provider Service specification.

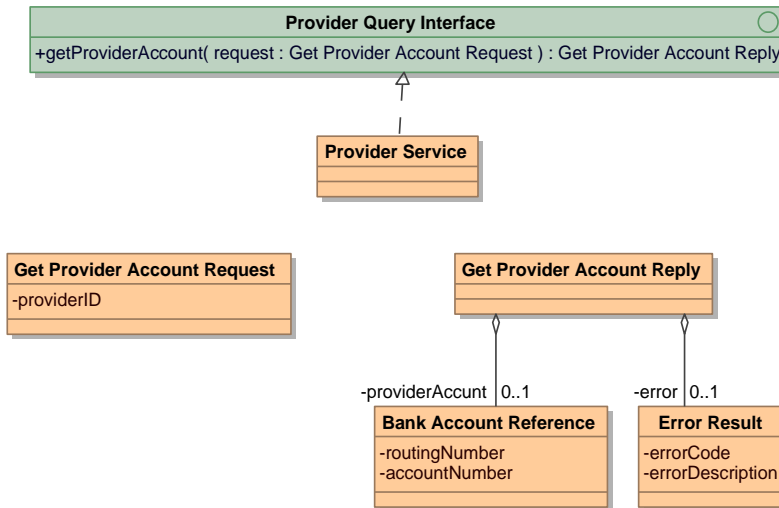


Figure 5-4
Provider Query Interface – getProviderAccount() Operation

6 Observable State

The observable state of the Payment Manager is shown in Figure 6-1. The diagram indicates which elements are cached copies of information owned by other services.

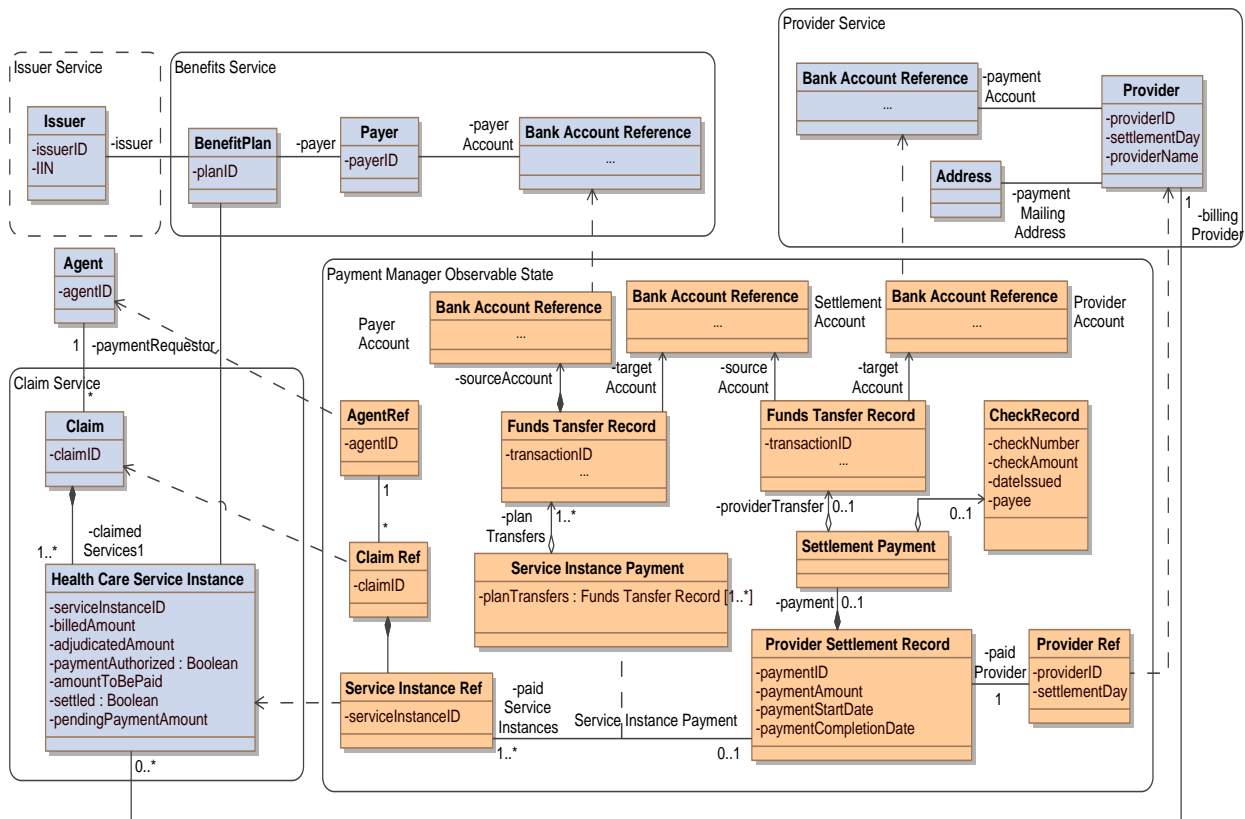


Figure 6-1
Payment Manager Observable State

7 Triggered Behaviors

There are two types of events that trigger behavior in the Payment Manager: the invocation of the `payClaim()` interface and the arrival of the time for settling deferred payments. There are two major variations in activity for the `payClaim()` invocation depending upon the value of the `immediatePayment()` flag.

7.1 Deferred Payment Behavior

The deferred payment behavior is shown in Figure 7-1. This behavior is triggered by the invocation of the Claim Payment Interface `payClaim()` operation. The behavior depends upon three services: the Benefits Service, the Banking Service, and the Claim Tracker. Errors in processing are indicated by the success flag being set to false and the presence of an Error Result in the returned data structure. By business rule, service instances that can be processed are completed, so it is possible to have some Service Instance Payment Results returned while others fail. In the event of failures, it is the responsibility of the service consumer (or whatever component is handling error recovery) to re-submit a payment request for the service instances that were not successfully completed.

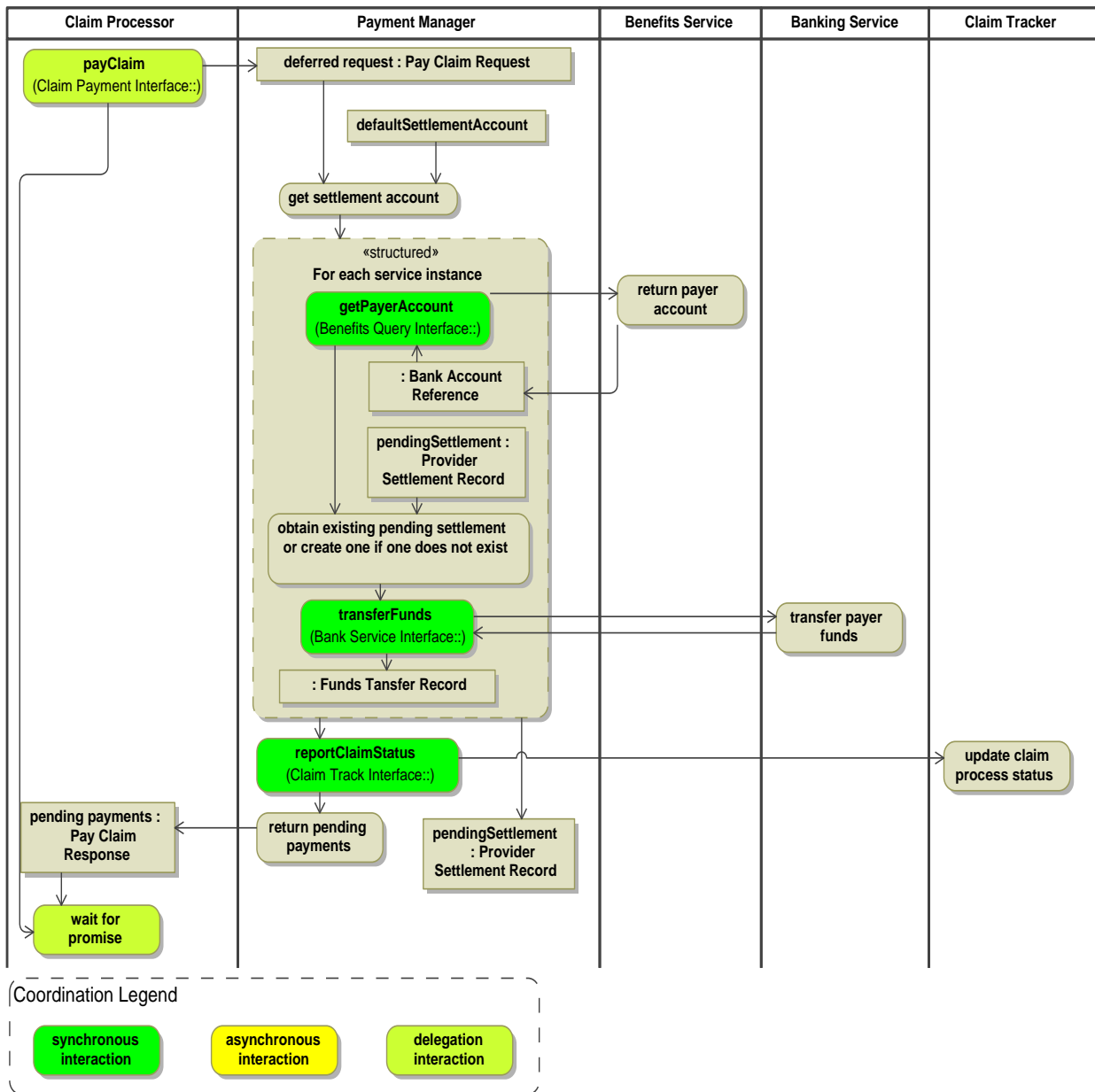


Figure 7-1
Deferred Payment Behavior

7.2 Settle Deferred Payments Behavior

The behavior for settling deferred payments is shown in Figure 7-2. The behavior is triggered when the time for settling with the provider is reached. At that time the pending settlements are retrieved. For each pending settlement, the provider account is retrieved and the funds are transferred from the settlement account to the provider account (Figure 5-2). Alternatively, if the provider is to be paid with a check, the banking service is directed to draw a check on the settlement account (Figure 5-2). This latter scenario variation is not shown in the picture. Finally, the claim status is updated (Figure 5-3).

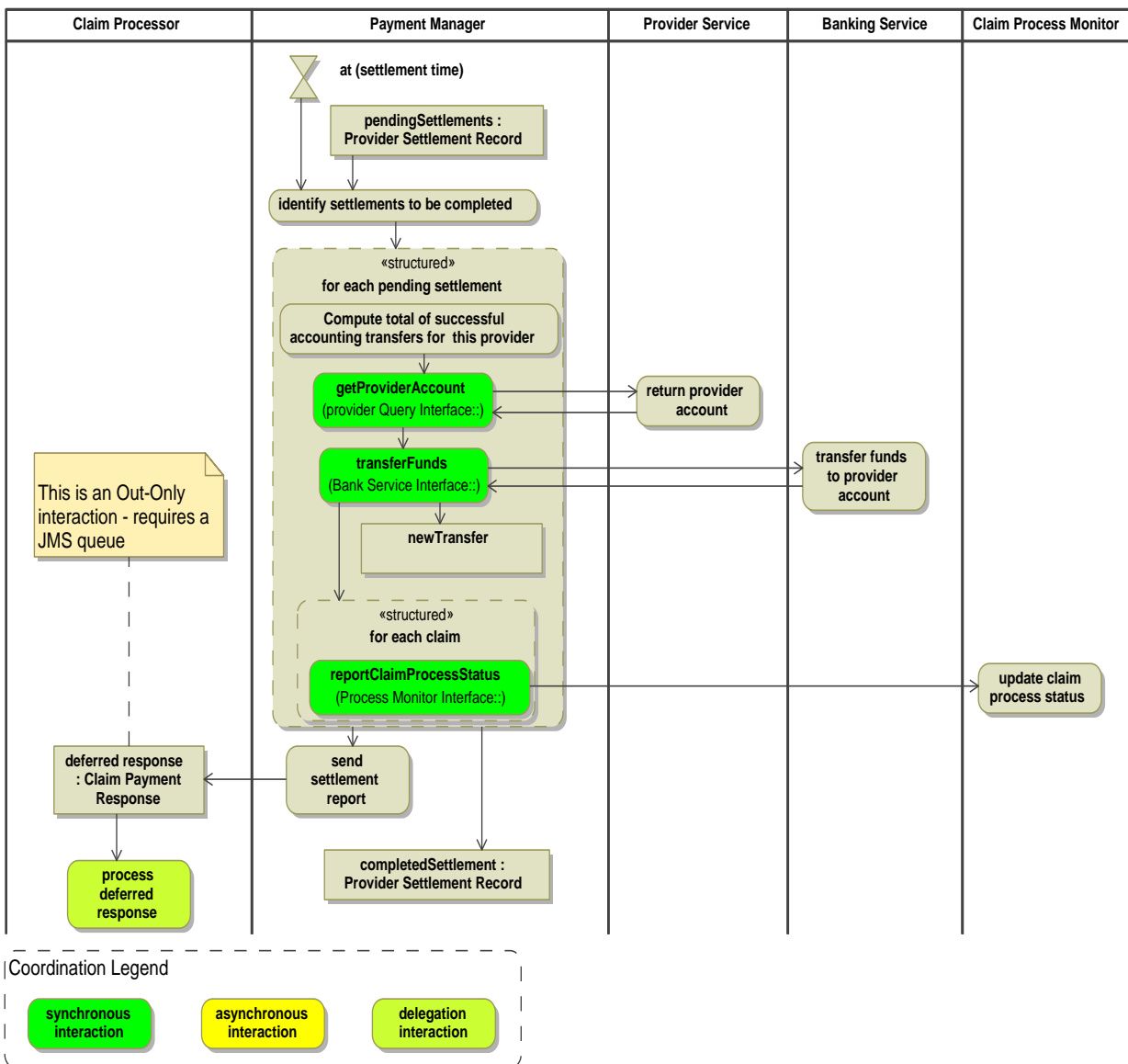


Figure 7-2
Settle Deferred Payments Behavior

7.3 Immediate Payment Behavior

The immediate payment behavior (Figure 7-3) is triggered by the invocation of the Claim Payment Interface payClaim() operation with the value of immediatePayment set to true. All of the interfaces used by this behavior have already been described.

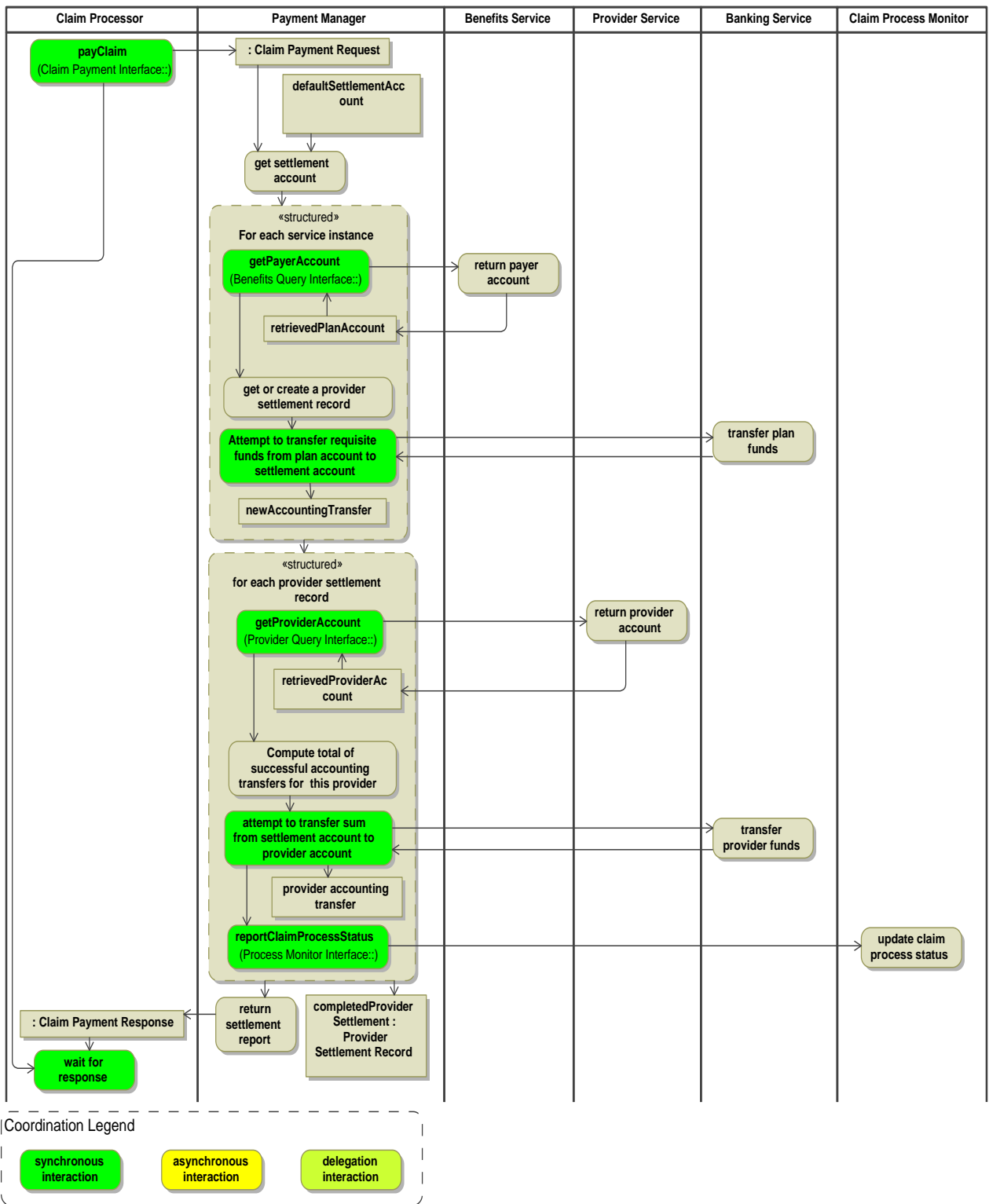


Figure 7-3
Immediate Payment Behavior

8 Coordination

From the service consumer's perspective, there are two coordination patterns available. The normal option, deferred payment, follows the Delegation with Confirmation pattern (Figure 3-1). The immediate payment option (Figure 3-2) is a synchronous request-reply interaction.

Most of the interactions with the back-end systems are synchronous request-reply. These include the Benefits Service, Provider Service, and Banking Service. The interactions with the Claim Tracker use asynchronous Out-Only coordination.

Since all interactions except those with the Claim Tracker return confirmations and response-time service-level agreements are in place for all these interfaces, breakdowns in these interactions can always be detected. No breakdown detection is available for claim status reporting to the Claim Tracker, which places a premium on the reliability of the communications infrastructure used for this communications. If a communication should be lost, the Claim Tracker will notice the absence of the event and report the process to be behind schedule.

9 Constraints

There are some restrictions on the interactions that can occur:

- [lb] It is invalid to call the Claim Payment Notification Interface `claimPaid()` operation for a claim for which the Claim Payment Interface's `payClaim()` operation has not been invoked.
- [lb] It is invalid to call the Claim Payment Interface `cancelPendingPayments` for a claim for which:
 - [lb] `payClaim()` has not been called
 - [lb] The payment has already been made

In a full specification, the triggered behavior mappings would include scenarios to indicate what would happen in each of these circumstances.

10 Non-Functional Behavior

10.1 Performance

Nouveau Health Care expects to handle up to 4.4 million claims per day. At peak times, `payClaim()` deferred payment requests may arrive at a rate of 620 requests/second. The service will provide a two second response time during these peak periods for average requests.

The average claim payment request has three service instances, but requests associated with hospital stays (about 1% of the total) may contain several hundred service instances. Response time for these requests will be 10 seconds.

Immediate payment requests account for about 1% of the total volume. Response time for an average request will be four seconds, and for a large request will be 60 seconds.

There are 1.4 million providers associated with Nouveau. Each provider is typically paid once a month on a working day. Thus the Settle Deferred Payments process runs about 67,000 times a day. Each execution will complete in 4 seconds, and the full batch will be completed in four hours.

At peak, the remaining Claim Payment Interface operations are each invoked 10 times/second and will provide a four second response time.

10.2 Availability within a Data Center

The payClaim() and claimPaid() operations will be available 99.999% of the time on a 24x7 basis. There will be no scheduled outage times for this operation. Maximum outage time per incident is 60 seconds.

The remaining Claim Payment Interface operations will be available 99.999% of the time from 6 AM through 12 PM Eastern time. Maximum outage time per incident is 60 seconds.

10.3 Site Disaster Recovery

In the event of a site disaster recovery, the recovery time objective for the Payment Manager is one hour, and the recovery point objective is 60 seconds.

10.4 Security

All invocations of the Claim Payment Interface operations require certificate-based authentication and authorization using web service standards. In all cases, WS-Security will be used to encrypt the message body.

11 Deployment

Claim Payment Interface operations will be accessible via SOAP over both HTTP and JMS. The endpoints for these operations are SOAP WSDL endpoints. The Claim Payment Notification Interface operation is an XML over JMS communication utilizing JMS queues. There is a separate queue for each agent submitting payClaim requests. The endpoints are as follows (*in the completed specification this table would be filled in*):

Table 11-1
Operation Endpoints

Operation	Development	Test	Production
payClaim			
claimPaid	<i>One queue per agent submitting claims</i>		
getProviderSettlement			
cancelPendingPayments			
getPendingSettlements			

Appendix A: Service Interface Specifications

Place WSDLs here

Appendix B: Referenced Interface Specifications

Place WSDLs here