



DCSA BOOKING CONFIRMATION - CUSTOMER ONBOARDING DOCUMENTATION

1. INTRODUCTION

The aim of this document is to provide the technical information required for customers and integration teams to consume **Booking Confirmation** messages through the Azure Portal.

The Booking Confirmation service provides booking-related information to subscribed customers through an **Azure Service Bus** integration. The service enables customers to receive booking confirmation information electronically and integrate the received messages into their own applications and business processes. The solution is designed to support the exchange of booking confirmation information between MSC systems and customer systems using standardized message formats and secure Azure-based integration components.

The document provides information about:

- Booking Confirmation events and message format
- Message properties and headers
- Booking filtering and subscription rules
- Azure Service Bus connectivity
- Available environments
- Message processing and error handling

2. BOOKING CONFIRMATION EVENTS AND FORMAT

The Booking Confirmation service provides booking-related events generated from the MSC booking process.

The events contain the information required by the customer to identify and process a booking confirmation, including booking references and other relevant booking details. The Booking Confirmation message is provided in **JSON format** and follows the agreed structure for the integration.

The message can contain information related to:

- Booking number
- eBooking number
- Booking status
- Booking confirmation details
- Customer and agency information
- Vessel and voyage information
- Port and routing information
- Equipment and container details
- Cargo information
- Other booking-related information applicable to the confirmation

The exact information available in the message depends on the booking data and the business scenario. The Booking Confirmation integration uses Azure-based components to process and deliver the message to the appropriate customer subscription. The Format of the Messages is compliant with the **DCSA BOOKING 2.0.3 standards**.

DCSA OpenAPI specification for the Booking process: [DCSA_BKG | 2.0.3 | dcsaorg | Studio](#)

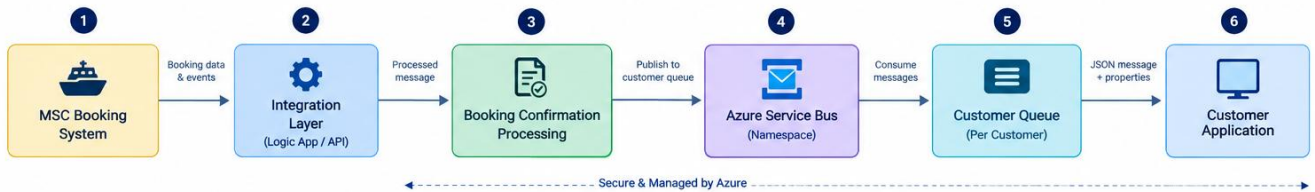
3. MESSAGE FLOW

The high-level Booking Confirmation message flow is:

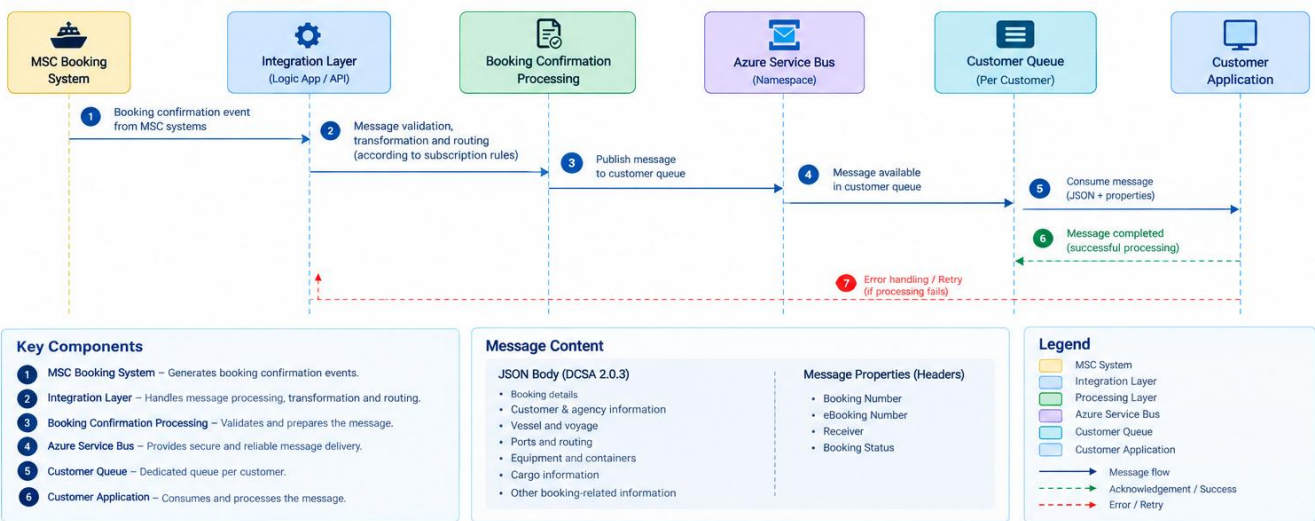
MSC Booking System → Integration Layer → Booking Confirmation Processing → Azure Service Bus → Customer Queue → Customer Application

Booking Confirmation – End-to-End Flow

This diagram illustrates the high-level flow of Booking Confirmation messages from the MSC systems to the customer application, including the Azure Service Bus integration and key components involved.



Detailed Message Flow (Sequence Diagram)



Once a booking confirmation is generated and the message satisfies the applicable subscription and filtering rules, the message is processed by the integration layer. The processed message is then published to the customer's dedicated Azure Service Bus queue. The customer application consumes the message from the queue and performs the required downstream processing. The Azure integration layer is responsible for message processing, transformation where applicable, routing, and delivery to the subscribed customer.

4. MESSAGE PROPERTIES

The Booking Confirmation message contains a JSON body along with message properties that provide additional information required for routing and processing. The properties can be used by the customer application to identify and process the message without having to inspect the complete JSON body.

The important message properties include:

4.1 Booking Number

The **Booking Number** identifies the booking associated with the Booking Confirmation message. This reference can be used by the consuming application to correlate the message with the corresponding booking in the customer's system.

4.2 eBooking Number

The **eBooking Number** is used as an additional booking reference where applicable. It can be used for customer-side filtering, identification, and correlation of booking-related messages.

4.3 Receiver

The **Receiver** property identifies the intended receiving party or routing destination. The receiver information is particularly important when multiple agencies, partners, or customer destinations are supported by the same integration flow.

4.4 Booking Status

The Booking Status provides information about the current state of the booking represented by the message.

5. FILTERING AND ROUTING RULES

The Booking Confirmation service uses filtering and routing rules to ensure that messages are delivered to the appropriate customer destination. The filtering logic is based primarily on booking-related information available in the message. Once a booking satisfies the configured customer subscription criteria, the corresponding Booking Confirmation message is routed to the customer's dedicated Azure Service Bus queue.

The filtering and routing configuration may include information such as:

- Customer
- Booking Number
- eBooking Number
- Contract or customer-specific reference
- Other agreed business criteria

This ensures that customers receive only the booking confirmation messages relevant to their subscription.

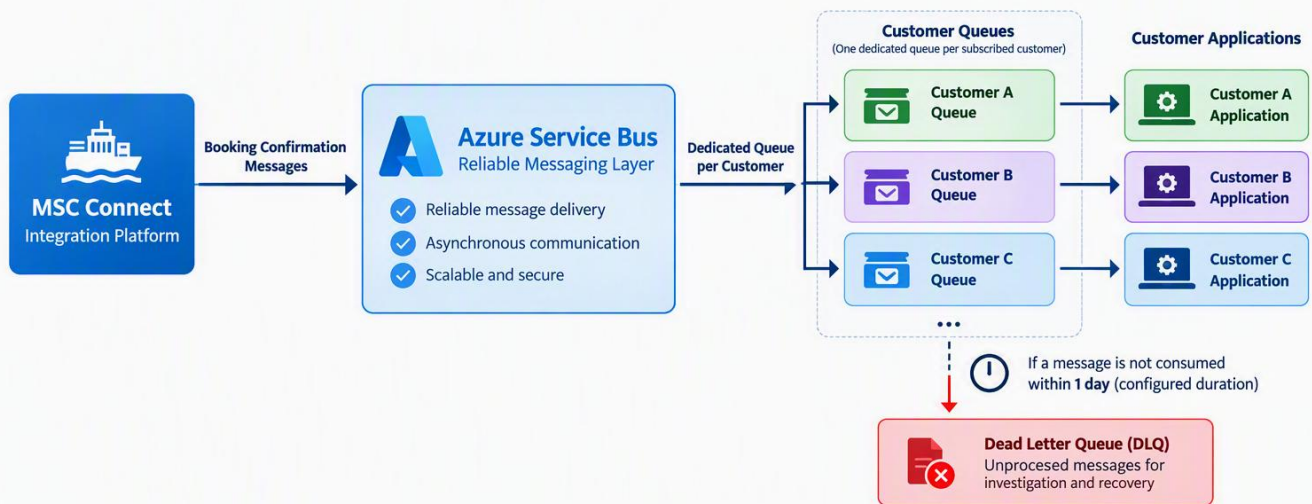
6. AZURE SERVICE BUS

The Booking Confirmation service uses **Azure Service Bus** for reliable message delivery between the MSC integration platform and the customer application. Each onboarded customer is provided with a dedicated queue or configured destination for consuming Booking Confirmation messages. The customer application is responsible for connecting to the Service Bus and consuming the messages from the assigned queue. The Azure Service Bus has a dedicated queue for every customer onboarded who has a subscription. There is a specific configuration that defines the amount of time for a message to be processed and read. This means that if messages are not **consumed within 1 day, they are moved to the dead letter recipient**. The Service Bus provides the messaging layer between the integration platform and the customer's consuming application.

Booking Confirmation Service

Message flow using Azure Service Bus

Reliable, secure and asynchronous delivery of Booking Confirmation messages from MSC Connect to customer applications.



Dedicated Queue

Each subscribed customer gets a dedicated queue.



Customer Responsibility

Customer applications connect to the assigned queue and consume the messages.



Message Retention

Messages must be consumed **within 1 day** (configurable).



Dead Letter Queue

Unconsumed messages are automatically moved to the DLQ for further investigation.

SECURE | RELIABLE | SCALABLE | BUSINESS CONTINUITY

POWERED BY MICROSOFT AZURE

7. MESSAGE PROCESSING

When a Booking Confirmation message is received by the customer application, the application should first validate the message and its required properties.

The recommended processing sequence is:

1. Receive the message from the Azure Service Bus queue.
2. Read the message properties.
3. Identify the Booking Number and other relevant references.
4. Validate the JSON message body.
5. Process the Booking Confirmation according to the customer's business requirements.
6. Successfully complete the Service Bus message after successful processing.
7. Handle failed messages according to the customer's retry and error-handling mechanism.

8. ERROR HANDLING AND RETRY

The customer application should implement appropriate error handling when consuming Booking Confirmation messages. If a message cannot be processed because of a temporary application or connectivity issue, the customer application should use an appropriate retry mechanism. If processing continues to fail, the message should be handled according to the Azure Service Bus dead-letter and retry configuration. Customers should maintain sufficient application logs to identify:

- Booking Number
- eBooking Number
- Message/Correlation reference
- Processing timestamp
- Processing status
- Error details

These details help the support and integration teams investigate message-processing issues.

9. ENVIRONMENTS

The Booking Confirmation integration is made available across the required environments for development, testing, and production activities.

The environments used for the integration include:

- **UAT**
- **PREPROD**
- **PRODUCTION**

UAT

The UAT environment is primarily used for customer connectivity and functional testing.

Customers can validate:

- Service Bus connectivity
- Authentication and access
- Message consumption
- JSON parsing
- Message property handling
- Booking Confirmation processing

PREPROD

The PREPROD environment is used for final integration validation before Production deployment. Customers can perform end-to-end testing and validate their application behaviour using the production-like integration setup.

PRODUCTION

The Production environment is used for live Booking Confirmation message processing. Production connection details and access information are provided as part of the customer onboarding process after the required setup and approvals are completed.

10. CONNECTION DETAILS

To consume Booking Confirmation messages, the customer must establish connectivity to the Azure Service Bus queue assigned during onboarding. The connection information provided to the customer includes the required Service Bus details, such as:

- Service Bus Namespace
- Queue Name
- Queue URL / Endpoint
- Authentication details
- Access policy or approved authentication mechanism

The customer should use the official Microsoft Azure Service Bus client libraries or the approved integration mechanism to connect to the queue. The connection credentials and access information must be handled securely and should not be exposed in source code, logs, or publicly accessible configuration files.

11. SECURITY AND ACCESS

Access to the Booking Confirmation Service Bus resources is controlled through Azure security and access mechanisms. Only authorized customer applications should be provided access to the dedicated queue.

Customers are responsible for:

- Securely storing authentication credentials
- Restricting access to authorized applications
- Preventing credentials from being committed to source-control repositories
- Rotating credentials where applicable
- Monitoring unauthorized access attempts and following their organization's security standards

The required access information will be shared with the customer through the agreed onboarding process.

12. CUSTOMER RESPONSIBILITIES

The customer is responsible for ensuring that their consuming application is ready to receive and process Booking Confirmation messages.

The customer should:

- Maintain a stable Service Bus connection
- Process messages within the configured retention period
- Implement retry and error-handling mechanisms
- Monitor message consumption
- Secure Service Bus credentials
- Notify the integration/support team of connectivity or processing issues

13. SUPPORT AND TROUBLESHOOTING

In case of an issue with Booking Confirmation message consumption, the customer should provide the available technical information to the support team.

The following information is recommended:

- Booking Number
- eBooking Number
- Approximate message timestamp
- Environment
- Queue name
- Error message
- Application logs
- Correlation or message reference, where available

Providing these details will help the integration team identify whether the issue is related to message generation, filtering, routing, Azure Service Bus delivery, connectivity, or customer-side processing.