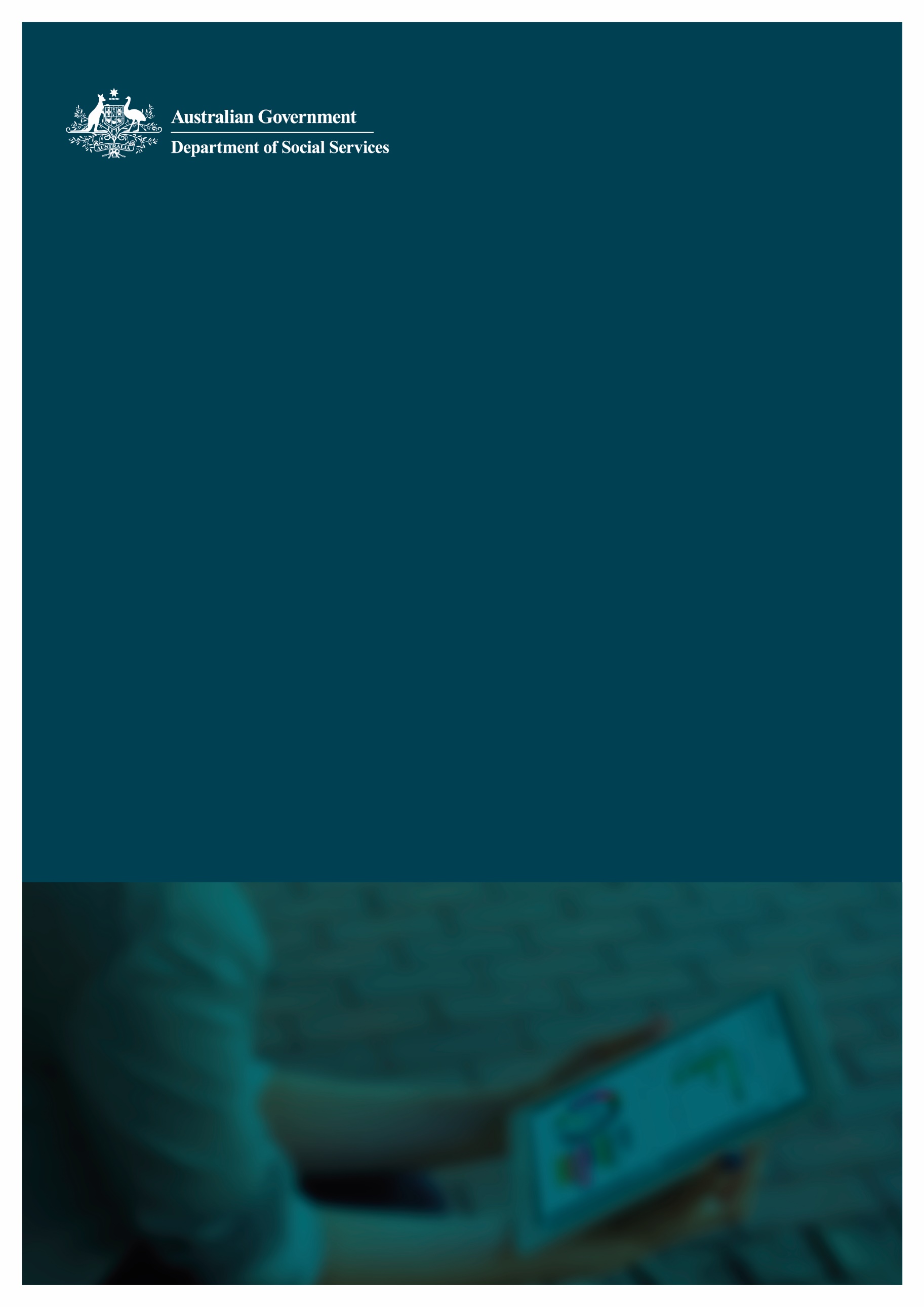
****Data Exchange System  
Web services

technical specifications

Version Number: 3

Revision Date: 30 June 2023

**Document change history**

| Version | Date | Change |
| --- | --- | --- |
| 1.0 | 14 October 2014 | Initial document release. First Ddocument reference 29723351 |
| 1.1 | 28 November 2014 | Remove CRN and ConsentGivenToDepartment as an element of Client as they are no longer required. |
| 1.2 | 21 July 2015 | Additions for ‘CHSP program specific’ changes for Client, Case & Session. |
| 1.3 | 15 October 2015 | Additions of new capability to manage organisation Outlet with following new methods   * GetOrganisationActivities * AddOutlet * UpdateOutlet * DeleteOutlet   Update to existing search client method   * SearchClient: new criteria item ‘Tags’.   Expected production release date is 12th Dec 2015 |
| 1.4 | 7 November 2016 | Changes to ‘Activity Specific Requirements’ |
| 1.5 | 25 July 2017 | Changes to ‘Activity Specific Requirements’.  Adding TopicCode for AddSession, UpdateSession and GetSession |
| 1.6 | 23 November 2017 | Correction for mandatory conditional and optional on ValidateForDuplicateClient |
| 1.7 | 23 April 2018 | IsHomeless (Boolean) changed to HomelessIndicatorCode (string) at DataCollection.Recipient operations.  Added EducationLevelCode, EmploymentStatusCode, IsClientACarer, and NDISEligibilityCode for DataCollection.Recipient operations.  Added ClientAttendanceProfileCode and EndDate for DataCollection.Case operations.  Added ServiceSettingCode for DataCollection.Session operations.  Added AssessedByCode for DataCollection.Assessments operations.  GetServiceTypeForOutletActivity method no longer supported. Now retrieved with GetOrganisationActivity method.  GetOrganisationActivity response now includes ActivitySpecificRequirements, ActivityServiceTypes, and AvailableAssessmentTypes.  IsAgdType removed from GetOutletActivities |
| 1.8 | 19 March 2019 | Add AvailableFrom and AvailableTo to:   * ServiceType, * ServiceTypeSpecificRequirements, * ActivitySpecificRequirements, * AvailableAssessmentType   for GetOrganisationActivities method |
| 1.9 | 1 September 2019 | Terminology updates (Service Provider to Organisation).  Further advice added to ClientId, CaseId, and SessionId validation rules.  Corrected order of address fields in ‘Get response’ examples. |
| 1.10 | 1 February 2021 | Added PropertyAgreementOutcome (and subsequently PropertyAgreementOutcomeCode, DateOfPropertyAgreement, and DidLegalPractitionerAssistInPropertyMediation) for DataCollection.Case operations. |
| 1.11 | 27 April 2021 | Added AgBusinessTypeCode for DataCollection.Case operations.  Added HardshipTypeCode, and ExternalReferralDestinationCode for DataCollection.Session operations.  Expected production release in June 2021. |
| 2 | 03 April 2023 | Added advice that client, case, and session Ids are not case sensitive.  General formatting changes. |
| 3 | 30 June 2023 | Addressed document readability and accessibility.  Formatting changes. |

Contents

[1 Overview 17](#_Toc136511055)

[1.1 The Data Exchange 17](#_Toc136511056)

[1.2 Purpose of this document 17](#_Toc136511057)

[2 High level data structure 18](#_Toc136511059)

[2.1 Data structure definition 19](#_Toc136511060)

[3 Interface availability 20](#_Toc136511062)

[3.1 Production 20](#_Toc136511063)

[3.2 Pre-production 20](#_Toc136511064)

[3.3 Interface addresses 20](#_Toc136511065)

[3.4 Updating period 21](#_Toc136511066)

[4 Service interface listing 22](#_Toc136511067)

[4.1 DataCollection.Utilities 22](#_Toc136511068)

[4.2 DataCollection.Reference 22](#_Toc136511069)

[4.3 DataCollection.Organisation 23](#_Toc136511070)

[4.4 DataCollection.Recipient 23](#_Toc136511071)

[4.5 DataCollection.Case 24](#_Toc136511073)

[4.6 DataCollection.Session 27](#_Toc136511074)

[4.7 DataCollection.Assessments 29](#_Toc136511075)

[4.8 DataCollection.Outlets 29](#_Toc136511076)

[5 Message Definitions 31](#_Toc136511077)

[5.1 Data Types 31](#_Toc136511078)

[5.2 Validation Rules 31](#_Toc136511079)

[5.3 Technical Requirements 31](#_Toc136511080)

[5.3.1 Compression 32](#_Toc136511081)

[5.3.2 Security 32](#_Toc136511082)

[5.3.3 Request and response definitions 32](#_Toc136511083)

[5.3.4 Request identifiers for web services 32](#_Toc136511084)

[5.3.5 Request schema 33](#_Toc136511085)

[5.3.6 Response schema 33](#_Toc136511086)

[5.3.7 ‘Header’ control 34](#_Toc136511087)

[5.3.8 Transaction data 34](#_Toc136511088)

[5.3.9 TransactionStatus 35](#_Toc136511089)

[5.3.9.1 TransactionStatus definition 35](#_Toc136511090)

[5.3.9.2 TransactionStatusCode descriptions 35](#_Toc136511091)

[5.3.9.3 Messages 36](#_Toc136511092)

[5.3.10 Error handling 36](#_Toc136511093)

[5.3.10.1 Schema and system errors. 36](#_Toc136511094)

[5.3.10.2 Schema error example 38](#_Toc136511095)

[5.3.10.3 System error example 39](#_Toc136511096)

[5.3.10.4 Business errors 40](#_Toc136511097)

[5.4 Statistical Linkage Key (SLK) 40](#_Toc136511099)

[5.4.1 SLK algorithm 40](#_Toc136511100)

[5.4.2 SLK regular expression 41](#_Toc136511101)

[5.4.3 SLK examples 42](#_Toc136511102)

[6 Interface method definitions 43](#_Toc136511103)

[6.1 DataCollection.Utilities 43](#_Toc136511104)

[6.1.1 Ping 43](#_Toc136511105)

[6.1.1.1 Ping request message definition 43](#_Toc136511106)

[6.1.1.2 Ping example request message 43](#_Toc136511107)

[6.1.1.3 Ping response message definition 44](#_Toc136511109)

[6.1.1.4 Ping example response message 44](#_Toc136511111)

[6.2 DataCollection.Reference 45](#_Toc136511112)

[6.2.1 GetReferenceData 45](#_Toc136511113)

[6.2.1.1 GetReferenceData request message definition 45](#_Toc136511114)

[6.2.1.2 GetReferenceData example request message 46](#_Toc136511115)

[6.2.1.3 GetReferenceData response message definition 48](#_Toc136511116)

[6.2.1.4 GetReferenceData example response message 48](#_Toc136511117)

[6.2.2 GetAssessmentReferenceDetails 49](#_Toc136511118)

[6.2.2.1 GetAssessmentReferenceData request message definition 50](#_Toc136511119)

[6.2.2.2 GetAssessmentReferenceDetails example request message 51](#_Toc136511120)

[6.2.2.3 GetAssessmentReferenceDetails response message definition 51](#_Toc136511122)

[6.2.2.4 GetAssessmentReferenceDetails example response message 53](#_Toc136511123)

[6.3 DataCollection.Organisation 56](#_Toc136511124)

[6.3.1 GetOrganisation 56](#_Toc136511125)

[6.3.1.1 GetOrganisation request message definition 56](#_Toc136511126)

[6.3.1.2 GetOrganisation example request message 56](#_Toc136511128)

[6.3.1.3 GetOrganisation response message definition 56](#_Toc136511130)

[6.3.1.4 GetOrganisation example response message 58](#_Toc136511132)

[6.3.2 GetOrganisationActivities 60](#_Toc136511133)

[6.3.2.1 GetOrganisationActivities request message definition 60](#_Toc136511135)

[6.3.2.2 GetOrganisationActivities example request message 60](#_Toc136511137)

[6.3.2.3 GetOrganisationActivities response message definition 60](#_Toc136511138)

[6.3.2.4 GetOrganisationActivities example response message 63](#_Toc136511140)

[6.3.3 GetOutlet 65](#_Toc136511141)

[6.3.3.1 GetOutlet request message definition 65](#_Toc136511143)

[6.3.3.2 GetOutlet example request message 66](#_Toc136511145)

[6.3.3.3 GetOutlet response message definition 66](#_Toc136511147)

[6.3.3.4 GetOutlet example response message 67](#_Toc136511148)

[6.3.4 GetOutletActivities 67](#_Toc136511149)

[6.3.4.1 GetOutletActivities request message definition 68](#_Toc136511151)

[6.3.4.2 GetOutletActivities example request message 68](#_Toc136511152)

[6.3.4.3 GetOutletActivities response message definition 68](#_Toc136511154)

[6.3.4.4 GetOutletActivities example response message 70](#_Toc136511155)

[6.3.5 GetUser 71](#_Toc136511156)

[6.3.5.1 GetUser request message definition 71](#_Toc136511158)

[6.3.5.2 GetUser example request message 71](#_Toc136511160)

[6.3.5.3 GetUser response message definition 71](#_Toc136511162)

[6.3.5.4 GetUser example response message 73](#_Toc136511164)

[6.4 DataCollection.Recipient 75](#_Toc136511165)

[6.4.1 AddClient 75](#_Toc136511166)

[6.4.1.1 AddClient request message definition 75](#_Toc136511168)

[6.4.1.2 AddClient example request message 79](#_Toc136511170)

[6.4.1.3 AddClient response message definition 82](#_Toc136511172)

[6.4.1.4 AddClient example response message 82](#_Toc136511174)

[6.4.2 GetClient 83](#_Toc136511175)

[6.4.2.1 GetClient request message definition 83](#_Toc136511177)

[6.4.2.2 GetClient example request message 83](#_Toc136511179)

[6.4.2.3 GetClient response message definition 84](#_Toc136511181)

[6.4.2.4 GetClient example response message 86](#_Toc136511183)

[6.4.3 SearchClient 88](#_Toc136511185)

[6.4.3.1 SearchClient request message definition 88](#_Toc136511186)

[6.4.3.2 SearchClient example request message 89](#_Toc136511188)

[6.4.3.3 SearchClient response message definition 90](#_Toc136511190)

[6.4.3.4 SearchClient example response message 91](#_Toc136511192)

[6.4.4 ValidateForDuplicateClient 92](#_Toc136511194)

[6.4.4.1 ValidateForDuplicateClient request message definition 93](#_Toc136511196)

[6.4.4.2 ValidateForDuplicateClient example request message 93](#_Toc136511198)

[6.4.4.3 ValidateForDuplicateClient response message definition 94](#_Toc136511200)

[6.4.4.4 ValidateForDuplicateClient example response message 95](#_Toc136511201)

[6.4.5 UpdateClient 97](#_Toc136511202)

[6.4.5.1 UpdateClient request message definition 99](#_Toc136511203)

[6.4.5.2 UpdateClient example request message 102](#_Toc136511204)

[6.4.5.3 UpdateClient response message definition 105](#_Toc136511206)

[6.4.5.4 UpdateClient response message 105](#_Toc136511208)

[6.4.6 DeleteClient 105](#_Toc136511210)

[6.4.6.1 DeleteClient request message definition 105](#_Toc136511212)

[6.4.6.2 DeleteClient example request message 106](#_Toc136511214)

[6.4.6.3 DeleteClient response message definition 106](#_Toc136511215)

[6.4.6.4 DeleteClient example response message 106](#_Toc136511217)

[6.5 DataCollection.Case 107](#_Toc136511218)

[6.5.1 AddCase 107](#_Toc136511219)

[6.5.1.1 AddCase request message definition 107](#_Toc136511221)

[6.5.1.2 AddCase example Rrequest message 109](#_Toc136511224)

[6.5.1.3 AddCase response message definition 111](#_Toc136511226)

[6.5.1.4 AddCase example response message 112](#_Toc136511228)

[6.5.2 GetCase 112](#_Toc136511230)

[6.5.2.1 GetCase request message definition 112](#_Toc136511231)

[6.5.2.2 GetCase Example Request Message 112](#_Toc136511232)

[6.5.2.3 GetCase response message definition 112](#_Toc136511233)

[6.5.2.4 GetCase example response message 114](#_Toc136511235)

[6.5.3 SearchCase 116](#_Toc136511237)

[6.5.3.1 SearchCase request message definition 116](#_Toc136511239)

[6.5.3.2 SearchCase example request message 117](#_Toc136511241)

[6.5.3.3 SearchCase response message definition 119](#_Toc136511243)

[6.5.3.4 SearchCase example response message 120](#_Toc136511245)

[6.5.4 UpdateCase 121](#_Toc136511247)

[6.5.4.1 UpdateCase example request definition 121](#_Toc136511249)

[6.5.4.2 UpdateCase example request message 123](#_Toc136511251)

[6.5.4.3 UpdateCase response message definition 125](#_Toc136511253)

[6.5.4.4 UpdateCase example response message 125](#_Toc136511255)

[6.5.5 DeleteCase 126](#_Toc136511257)

[6.5.5.1 DeleteCase request message definition 126](#_Toc136511259)

[6.5.5.2 DeleteCase example request message 126](#_Toc136511260)

[6.5.5.3 DeleteCase response message definition 126](#_Toc136511261)

[6.5.5.4 DeleteCase example response message 126](#_Toc136511263)

[6.6 DataCollection.Session 127](#_Toc136511264)

[6.6.1 AddSession 127](#_Toc136511265)

[6.6.1.1 AddSession request message definition 127](#_Toc136511267)

[6.6.1.2 AddSession example request message 129](#_Toc136511269)

[6.6.1.3 AddSession response message definition 131](#_Toc136511271)

[6.6.1.4 AddSession example response message 132](#_Toc136511273)

[6.6.2 GetSession 132](#_Toc136511275)

[6.6.2.1 GetSession request message definition 132](#_Toc136511277)

[6.6.2.2 GetSession example request message 132](#_Toc136511279)

[6.6.2.3 GetSession response message definition 133](#_Toc136511281)

[6.6.2.4 GetSession example response message 135](#_Toc136511282)

[6.6.3 UpdateSession 138](#_Toc136511283)

[6.6.3.1 UpdateSession request message definition 138](#_Toc136511285)

[6.6.3.2 UpdateSession example request message 140](#_Toc136511287)

[6.6.3.3 UpdateSession response message definition 142](#_Toc136511289)

[6.6.3.4 UpdateSession example response message 142](#_Toc136511291)

[6.6.4 DeleteSession 143](#_Toc136511292)

[6.6.4.1 DeleteSession request message definition 143](#_Toc136511294)

[6.6.4.2 DeleteSession example request message 143](#_Toc136511296)

[6.6.4.3 DeleteSession response message definition 143](#_Toc136511298)

[6.6.4.4 DeleteSession example response message 143](#_Toc136511300)

[6.7 DataCollection.Assessments 144](#_Toc136511302)

[6.7.1 UpdateClientAssessments 144](#_Toc136511303)

[6.7.1.1 UpdateClientAssessments request message definition 144](#_Toc136511305)

[6.7.1.2 UpdateClientAssessments example request message 145](#_Toc136511307)

[6.7.1.3 UpdateClientAssessments response message definition 146](#_Toc136511309)

[6.7.1.4 UpdateClientAssessments example response message 146](#_Toc136511311)

[6.7.2 UpdateSessionAssessments 146](#_Toc136511313)

[6.7.2.1 UpdateSessionAssessments request message definition 147](#_Toc136511315)

[6.7.2.2 UpdateSessionAssessments example request message 148](#_Toc136511317)

[6.7.2.3 UpdateSessionAssessments response message definition 148](#_Toc136511319)

[6.7.2.4 UpdateSessionAssessments example response message 149](#_Toc136511321)

[6.8 DataCollection.Outlets 150](#_Toc136511322)

[6.8.1 AddOutlet 150](#_Toc136511323)

[6.8.1.1 AddOutlet Request Message Definition 150](#_Toc136511325)

[6.8.1.2 AddOutlet example request message 151](#_Toc136511327)

[6.8.1.3 AddOutlet response message definition 152](#_Toc136511329)

[6.8.1.4 AddOutlet Example response message 152](#_Toc136511331)

[6.8.2 UpdateOutlet 152](#_Toc136511332)

[6.8.2.1 UpdateOutlet request message definition 153](#_Toc136511334)

[6.8.2.2 UpdateOutlet example request message 154](#_Toc136511336)

[6.8.2.3 UpdateOutlet response message definition 154](#_Toc136511338)

[6.8.2.4 UpdateOutlet example response message 155](#_Toc136511340)

[6.8.3 DeleteOutlet 155](#_Toc136511342)

[6.8.3.1 DeleteOutlet Request message definition 155](#_Toc136511344)

[6.8.3.2 DeleteOutlet example request message 155](#_Toc136511346)

[6.8.3.3 DeleteOutlet response message definition 155](#_Toc136511348)

[6.8.3.4 DeleteOutlet example response message 156](#_Toc136511350)

[6.9 Common message definitions 156](#_Toc136511351)

[6.9.1 Request message definitions 156](#_Toc136511352)

[6.9.1.1 Search request message definition 156](#_Toc136511353)

[6.9.2 Response message definitions 157](#_Toc136511354)

[6.9.2.1 Client response message definition 157](#_Toc136511355)

[7 Appendix A. Activity specific requirements 160](#_Toc136511356)

[8 Appendix B. Service type specific requirements 162](#_Toc136511357)

# Overview

## The Data Exchange

The Data Exchange represents the program performance reporting approach for client facing funding agreements. The Department of Social Services (DSS) progressively introduced standardised, prioritised, and collaborative reporting processes across grants programs from 1 July 2014.

This approach to reporting is a streamlined, processes automated and shifts the focus of performance measurement from outputs to more meaningful information about service delivery outcomes.

The requirements of the Data Exchange are divided into two parts: a small set of mandatory priority requirements that all organisations report, and an extended data set, known as the partnership approach. As part of the partnership approach, organisations are encouraged to collect the Standard Client/Community Outcome Reporting (SCORE) information for as many clients as practical.

Organisation can access a free IT system to manually input client data. This helps record clients, service and outcomes data that meet funding agreement performance data requirements and allows organisations to confidentially manage their core client and case information.

## Purpose of this document

This document specifies the technical requirements for organisations who would prefer to transmit their data using system-to-system transfers. This will be achieved by transferring data from a third party software application using the Data Exchange System Web Services interface.

The following is described within this document:

1. Interface availability,
2. Service interface listing,
3. Message definitions, and
4. Interface method definitions.

You should read this document in conjunction with the [Data Exchange Protocols](https://dex.dss.gov.au/data-exchange-protocols), available on the [Data Exchange Website](https://dex.dss.gov.au/).

For assistance regarding content in this document, please contact the helpdesk by email at [DSSDataExchange.Helpdesk@dss.gov.au](mailto:DSSDataExchange.Helpdesk@dss.gov.au) or on 1800 020 283.

# High level data structure

The following diagram illustrates a high-level relationship between the entities.

This diagram illustrates a high level relationship between the entities. The entities are User, Organisation, Activity, Outlet, Outlet Activity, Case, Case Client, Client, Session, Session Client, Session Assessment and Client Assessment.

## Data structure definition

| No. | Data | Definition |
| --- | --- | --- |
|  | Activity | An **activity** is the program that an organisation is funded to deliver. |
|  | Client | A **client** is an individual who receives a service as part of a funded activity. |
|  | Client assessment | A **client assessment** is an assessment (e.g. client circumstances, client goal or client satisfaction assessment) conducted against clients who have attended a session. |
|  | Case | A **case** is a grouping of one or more sessions. A case may consist of data that relates to all activities and/or specific activities. |
|  | Case client | A **case client** representsa client who is involved with the case. Additional data may be recorded against clients associated with a case. |
|  | Organisation | An organisation is a funded service provider. |
|  | Outlet | An **outlet** is the location of where services are being delivered from, as referenced in the funding agreement. |
|  | Outlet activity | An **outlet activity** represents an outlet and the activity that outlet is able to deliver. |
|  | Session | A **session** is an instance or episode of service delivered to a client or a group of clients. A session may consist of data that relates to all activities, specific activities, and/or specific activities and service types. |
|  | Session client | A **session client** is a client who is involved with the case **and** has attended a particular session. |
|  | Session assessment | A **session assessment** is a group assessment (e.g. community assessment) conducted against all clients who have attended a session. |
|  | User | A **user** is a representative of the organisation. |

# Interface availability

The following section provides information on interface availability. Software designers should take into account availability of the Data Exchange system when designing their software.

## Production

It is expected that the interface will be generally available[[1]](#footnote-1) at all times. In the event that the interface is unavailable, DSS will endeavour to provide information to the organisations about the outage. For support, please visit the [Data Exchange helpdesk page](https://dex.dss.gov.au/helpdesk).

Software designers should be aware of the availability of the Data Exchange system when designing their software. If the interface is unavailable, it is expected that the Service Provider Software will store the data and when the interface is available, the data will be re‑sent. For example, the software may try to send the data once it has been entered, however if the interface is unavailable it should be stored and resent again later. Alternatively, the Service Provider Software may batch up the data and send this through when the interface is available again. Therefore, the unavailability of the Data Exchange System interface should not stop the day-to-day operation of the organisation.

A ‘Ping’ interface has been defined that will enable software to query the Data Exchange system and can be used to test connectivity and authentication. The interface provides no functionality other than to return a blank response. The intent of the interface is to assist in testing and diagnosing connection and authentication issues.

Ideally, DSS would prefer that software be designed so that organisations using the same software do not interface data at exactly the same time. That is, DSS would prefer the load of data from a software provider to be ‘randomly’ spread across the day and not all transmitted at the same time.

## Pre-production

Any pre-production system provided by DSS, for testing purposes, will have varying levels of availability. Software providers should contact the Data Exchange helpdesk to determine availability before using a pre-production system.

## Interface addresses

The Data Exchange System Services will be available at the following URLs.

| Environment | Address |
| --- | --- |
| Production | https://api.dss.gov.au/datacollection/dex |
| Pre-Production | https://staging-api.dss.gov.au/datacollection/dex |

## Updating period

Input into the Data Exchange system will only be allowed to be added or modified within a fixed updating period.

# Service interface listing

The following sections provide a listing and description of the interfaces that comprise the Data Exchange System Web Service Interface. **All methods are for a single organisation associated with the web service user**.

## DataCollection.Utilities

| Operation | Date Locking | Description |
| --- | --- | --- |
| Ping | Not applicable | The intent of this method is to enable organisations to test connectivity and authentication with the Data Exchange system. |

## DataCollection.Reference

| Operation | Date Locking | Description |
| --- | --- | --- |
| GetReferenceData | Not applicable | Generic service to Get a list requested ReferenceData.  The following reference data are downloadable via this interface and returns a list of active code, description and order.  Current reference codes:   1. All 2. AboriginalOrTorresStraitIslanderOrigin 3. AccommodationType 4. AgBusinessType 5. Ancestry 6. AssessedBy 7. AssessmentPhase 8. AttendanceProfile 9. Country 10. Disability 11. DVACardStatus 12. EducationLevel 13. EmploymentStatus 14. ExitReason 15. ExternalReferralDestination 16. ExtraItem 17. Gender 18. HardshipType 19. HomelessIndicator 20. HouseholdComposition 21. IncomeFrequency 22. Language 23. MainSourceOfIncome 24. MigrationVisaCategory 25. MoneyBusinessCommunityEducationWorkshop 26. NDISEligibility 27. ParentingAgreement 28. ParticipationType 29. PropertyAgreement 30. ReasonForAssistance 31. ReferralPurpose 32. ReferralSource 33. ReferralType 34. State 35. ScoreType 36. Section60ICertificateType 37. ServiceSetting 38. Topic |
| GetAssessmentReferenceDetails | Not applicable | Service to Get a list requested AssessmentReferenceDetails.  Please use GetOrganisationActivities for AvailableAssessmentTypes related to activity you are working with. |

## DataCollection.Organisation

| Operation | Date Locking | Description |
| --- | --- | --- |
| GetOrganisation |  | Gets the user’s organisation. |
| GetOrganisationActivities |  | Gets the user’s organisation activities. |
| GetOutlet |  | Gets the user’s outlet for the given outlet id. |
| GetOutletActivities |  | Gets user’s outlet activities. |

## DataCollection.Recipient

| Operation | Date Locking | Description |
| --- | --- | --- |
| AddClient |  | Adds a Client with the following given details:   * ClientId (unique organisation’s client Id) * Consent * Personal information * Disabilities |
| GetClient |  | Gets the specified client with consent, personal information, disabilities, cases. |
| SearchClient |  | Search client records which match the specified search criteria. |
| ValidateForDuplicateClient |  | Gets the duplicate clients which match the specified criteria. |
| UpdateClient |  | Updates the specified client with the following given details:   * Consent * Personal information * Disabilities |
| DeleteClient |  | Deletes the client that matches the specified ClientId (organisation’s client Id). |

## DataCollection.Case

| Operation | Date Locking | Description |
| --- | --- | --- |
| AddCase |  | Adds a case with the following given details:   * CaseId (unique organisation’s case Id) * OutletActivityId * No of unidentified clients * ClientAttendanceProfileCode * End date * Ag Business Type Code * Case client details   + Identified client Ids (organisation’s client Ids)   + Reasons for assistance     - Assistance needed code     - IsPrimary   + Referral source code   + Exit reason code * Parenting agreement outcome   + Parenting agreement outcome code   + Date of parenting agreement   + Did legal practitioner assist with formalising agreement * Section60I   + Section 60I Certificate Code   + Date of Section 60I agreement * Property Agreement Outcome   + Property agreement outcome code   + Date of property agreement   + Did legal practitioner assist In property mediation |
| GetCase |  | Gets the specified case details along with the attached identified clients and sessions. |
| SearchCase |  | Search case records which match the specified search criteria. |
| UpdateCase |  | Updates the specified case with the following given details:   * OutletActivityId * No of unidentified clients * ClientAttendanceProfileCode * End date * Ag business type Code * Case client details   + Identified client Ids (organisation’s client Ids)   + Reasons for assistance     - Assistance needed code     - IsPrimary   + Referral source code   + Exit reason code * Parenting agreement outcome   + Parenting agreement outcome code   + Date of parenting agreement   + Did legal practitioner assist with formalising agreement * Section60I   + Section 60I Certificate Code   + Date of Section 60I Agreement * Property agreement outcome   + Property agreement outcome code   + Date of property agreement   + Did legal practitioner assist In property mediation |
| DeleteCase |  | Deletes the specified case. |

## DataCollection.Session

| Operation | Date Locking | Description |
| --- | --- | --- |
| AddSession |  | Adds a session to the specified case with the following given details:   * CaseId (unique organisation’s case Id) * Session Id (unique organisation’s session Id) * Session date * Service type Id * No of unidentified clients * Fees charged * Money business community education workshop code * Interpreter present * Session client details   + Identified client Ids (organisation’s client Ids)   + Participation code   + Client referral out with purpose * Time in minutes * Total cost * Topic * Quantity * Extra items * ServiceSettingCode * Hardship type code * External Referral Destination code |
| GetSession |  | Gets the specified session details along with session clients. |
| DeleteSession |  | Deletes a specified session. |
| UpdateSession |  | Updates the following given details for specified session:   * CaseId (unique organisation’s vase Id) * Session Id (unique organisation’s dession Id) * Session fate * Service type Id * No of unidentified vlients * Fees charged * Money business community education workshop code * Interpreter present * Session client details   + Identified client Ids (organisation’s client Ids)   + Participation code   + Client referral out with purpose * Time in minutes * Total cost * Topic * Quantity * Extra items * ServiceSettingCode * Hardship type code * External referral destination code |

## DataCollection.Assessments

| Operation | Date Locking | Description |
| --- | --- | --- |
| UpdateClientAsssements |  | Adds or updates the following given details for specified client in a session:   * Case Id (organisation’s case Id) * Session Id (organisation’s session Id within a case) * Client Id (organisation’s client Id associated with the case) * Assessments and scores |
| UpdateSessionAssessments |  | Adds or updates the following given details for specified session in a case:   * Case Id (organisation’s case Id) * Session Id (organisation’s Session Id within a case) * Assessments and SCOREs |

## DataCollection.Outlets

| Operation | Date Locking | Description |
| --- | --- | --- |
| AddOutlet |  | Adds an Outlet with the following details:   * Outlet name * Address   + Suburb   + State   + Postcode   + Address line 1   + Address line 2 * OutletActivities   + OrganisationActivityId   + Start date   + End date |
| UpdateOutlet |  | Updates an Outlet specified by Outlet Id with the following details:   * Outlet Id * Outlet name * Address   + Suburb   + State   + Postcode   + Address Line 1   + Address Line 2 * OutletActivities   + OrganisationActivityId   + Start Date   + End Date |
| DeleteOutlet |  | Deletes an Outlet if no case data is associated with it. Outlet activities are deleted also if no case data is associated with them. If case data is associated with an outlet activity then it will be end-dated to the current date:   * Outlet Id |

# Message Definitions

The following sections detail the various elements of the messages within the interfaces provided by the Data Exchange System Web Service Interface.

## Data Types

Definitions of the data types used in the tables below can be found at the World Wide Web Consortium website. The URL for the specifications for the data types is:

http://www.w3.org/TR/xmlschema-2/#built-in-datatypes

## Validation Rules

The Message Definition tables in the following sections contain a column with Validation Rules. These are the rules that will be applied to the data as it is processed. Some of the data elements have a validation rule like ‘Must be a code provided in the Language reference table’.

There will be times when a reference date will be specified next to the rule to a rule. If so, it would be to make sure the data element is valid at a particular point in time. The point in time being checked against will be properly specified against the element to that it applies.

## Technical Requirements

Most DSS Web services will cater for requests using SOAP 1.2. This is because using SOAP provides XML schema support for more complex data types. The transport method supported is SSL-encrypted HTTP. Web service calls using HTTP-GET and HTTP-POST are not currently supported (this might change in the future as REST Web Services become more widely supported) and not documented in this standard.

In order to call the DSS Web services, an application or SDK capable of calling XML web services is required. The application or SDK must support the following:

* XML 1.0;
* SOAP 1.2;
* HTTP 1.1; and
* WSE Authentication Headers. (version 3.0)
* Standard Authentication Headers
* Custom Headers

When implementing a responsive application, bandwidth requirements must be taken into consideration. Bandwidth requirements depend on many factors. These include:

* Size of the payload sent to the Data Exchange for processing;
* Frequency of the requests; and
* Data compression.

DSS web services have been designed to minimise the network traffic payload as much as possible. The services provided by DSS are therefore not bandwidth intensive. However, to ensure best performance, DSS recommends a broadband connection for both upstream and downstream traffic. Minimum bandwidth recommendations will be confirmed during performance testing.

### Compression

Compression is currently not supported by DSS Web Services.

### Security

DSS Web services will use industry standard WSE Authentication Headers (UserName token).

The following security rules apply to login passwords for DSS Web services:

* Web Service Passwords will not expire;
* Web Service UserIDs do not have access to online systems;
* Strong passwords will be used – They must contain a combination of upper and lowercase characters, numbers and special characters (e.g. #, @, $);
* Password must be between 10 and 15 characters long;
* The maximum number of failed logon attempts before the account will be locked is 5;
* If the account is locked, [DSSDataExchange.Helpdesk@dss.gov.au](mailto:DSSDataExchange.Helpdesk@dss.gov.au) must be contacted to reset the password;

### Request and response definitions

The following object types are used generally in all DSS Web services.

* Response objects and request objects;
* ‘Header’ control structure (soap header)
* Response transaction data and request transaction data.

### Request identifiers for web services

Request identifiers (**ClientRequestIDs**) are an important part of DSS Web Services. They are included in every batch web method call (sent as part of the RequestControl Structure described in the next section) to identify an individual request. The length of the **ClientRequestID** must be between 1 and 36 characters, and can be of any format. An obvious choice is to use a Globally Unique Identifier (GUID), but using an integer starting from 0 and incrementing by 1 with each web method call is also valid.

The value of the **ClientRequestID** must only be unique within an external organisation or stakeholder and it must retain uniqueness over time. It is the responsibility external organisation or stakeholder to ensure that all its client systems internally keep track of all previous **ClientRequestID (s)** and to synchronise generation of unique **ClientRequestID****(s)** for new calls.

Every time a batch method is called, a new **ClientRequestIDs** must be sent to the server. When returning the processing results of the request, the server response will include the same **ClientRequestID** (as part of the Response Header’ control structure described in the next section). In order to guarantee that the server does not process the same request twice due to a communication failure, calls to a web method providing an old **ClientRequestIDs** will result in a *Soap Fault* due to ‘Duplicate’ request and no new processing will be performed.

### Request schema

The request schema for DSS Web Services holds all the information a client needs to call a DSS Web service method. It consists of request control data (one **‘Header’ Control**) sent as SOAP Header, plus request transaction data elements (one or more). If the request is being sent for a real-time method, this ‘Header’ is optional and only one transaction data element is allowed.

This diagram shows how a request flows through the DSS Data Exchange.  The request schema for DSS Web Services holds all the information a client needs to call a DSS Web service method. It consists of request control data (one “Header” Control) sent as SOAP Header, plus request transaction data elements (one or more).
The SOAP Header includes create date time, source ID, target ID and Client request ID.
The Request transaction data includes the request structure or request field.

**Figure 1: General Request Diagram**

### Response schema

Similarly, the response schema is a container for all information a DSS Web service sends to a client. It contains response control data (one **‘Header’ control**) sent as soap header if the request provided a ‘Header’, plus one response element containing the return transaction data elements.

This diagram shows how a response flows through the DSS Data Exchange. The response schema is a container for all information a DSS Web service sends to a client. It contains response control data (one “Header” control) sent as Soap Header if the request provided a “Header”, plus one response element containing the return transaction data elements.
The SOAP Header includes Create date time, Source ID, Target ID and Client Request ID.
The Response Transaction Data includes a response field for the transaction status.
The transaction status code can be Success, Warning, Failed and the transaction status message includes Debug, Informaiton, Warning, Error.

**Figure 2: General response schema diagram**

### ‘Header’ control

**Header** is a container for all request/response control data. The **Header** control contains the following fields:

| Field Name | XML Type | Description |
| --- | --- | --- |
| CreateDateTime | dateTime | Date and time on the client at the time of the call. |
| SourceID | string | The client’s system ID. (Optional) |
| TargetID | string | The target system ID. (Optional) |
| ClientRequestID | string | The client request identifier. Max 36 chars. |

### Transaction data

Transaction data contains business information specific to the method being called. Request transaction data contains data required to perform the required business service, while response transaction data contains the results of processing the request. For example, in a method that returns client information (i.e. search); request transaction data will contain the client input details, while response transaction data will contain client detail information that matches the input search details.

Because each method has different input and output fields, each method defines different request and response transaction data schemas. There is however, a field that is common across transaction schemas: **TransactionStatus**.

### TransactionStatus

**TransactionStatus** is an element common only to response transaction schemas. It provides information about the results of business processing for an individual transaction. The **TransactionStatus** structure always has two fields: **TransactionStatusCode** and **Messages**. **TransactionStatusCode** indicates the processing status of the transaction, and **Messages** is an array of **Message** elements. Within the context of **TransactionStatus**, each **Message** element holds business validation or business error messages for that transaction.

#### TransactionStatus definition

| Field | Mandatory Conditional Optional | Data Type | Length | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatusCode | Mandatory | Enumeration |  | See below. |
| Messages | Mandatory |  |  | A collection of Message elements. |
| Messages.Message |  |  |  | This element provides extended information on the status of the request. There may be no messages. |
| Message.MessageId | Mandatory | Integer |  | Identifies the message. |
| Message.MessageDescription | Mandatory | String |  | The text of the message. |
| Message.MessageLevel | Mandatory | Enumeration |  | Describes the type of message. Valid values are **Error**, **Information**, **Warning and Debug**. See below. |

#### TransactionStatusCode descriptions

| TransactionStatusCode | Description |
| --- | --- |
| SUCCESS | The individual transaction processed correctly according to its respective business rules. Other response transaction data fields will contain valid output. |
| FAILED | The individual transaction failed processing according to business rules. Other response transaction data fields will not contain valid output. |
| WARNING | The individual transaction processed correctly but with warnings. Other response transaction data fields will contain valid output; however, the messages element will contain warning messages that should be investigated by the client in order to maintain data and processing integrity. |

#### Messages

The **TransactionStatus** fields contain the *Messages* element. This element is used to hold any messages concerned with either the request or its constituent transactions. The **Messages** element itself only contains an array of individual **Message** elements. Each **Message** element contains three fields: **MessageId**, **MessageDescription** and **MessageLevel**. The **MessageId** and **MessageDescription** elements will be listed by the individual DSS system.

| MessageSeverity | Description |
| --- | --- |
| DEBUG | The message is debug message. Activated only on special cases. |
| INFORMATION | The message is of an informational nature. |
| WARNING | The message provides a warning that should be investigated. When a warning message is produced, it usually implies that the request or transaction was successful, but with caveats. |
| ERROR | The message is produced when the request or transaction has failed. It contains reasons as to why it failed and what caused the error. |

### Error handling

#### Schema and system errors.

A request received by a DSS Web service is validated against the respective schema for that method. The validation performs checks to do with the format and structure of the request fields. If the request does not comply with the schema, the returning SOAP message will not contain any response data, only a SOAP fault.

Security and System errors will also be returned as a SOAP fault. A Security error is when a client attempts to call a DSS service with invalid credentials, or the user does not have enough access to the service.

A system error is any other condition, other than business errors or schema errors. For example, sending a **ClientRequestID** on a request, which has already been used in the past, or errors in the request control data (**Header Control**).

For all SOAP Faults, DSS will place, within the *detail* element, *a* **Fault** element, which is the parent for all soap fault errors for all DSS Services.

The**Fault** element contains:

| Field Name | XML Type | Description |
| --- | --- | --- |
| ReferenceID | string | A unique reference of the fault instance (metadata/MessageID). |
| FaultCode | string | A unique reference of the error. |
| Category | string | A unique reference of the error. |
| FaultText | normalizedString | A friendly error message of the error that occurs. The audience of this error message is the Tier-2 operation team. |
| RawFaultData | anyType | The full error stack of the error including nested error. |
| InstanceID | string | Internal to DSS: A unique reference of the Fusion Middleware instance in the cluster. For BPEL process, it will be set to the BPEL instance ID. For organisation errors, this field will be left blank. |

**Category** can take on the following values:

| MessageSeverity | Description |
| --- | --- |
| Timeout | The call has timed-out. |
| Unreachable | A non-existing service has been called. |
| ValidationError | The message provides a warning that should be investigated. When a warning message is produced, it usually implies that the request or transaction was successful, but with caveats. |
| UnknownOperation | A non-existing method within a service has been called. |
| InternalServerError | Error within service. |
| Other | Unknown error. |

#### Schema error example

An example of a server response to a request, which does not pass schema validation look like:

<?xml version="1.0" encoding="utf-8" standalone="yes" ?>

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<soap:Fault>

<faultcode>soap:Client</faultcode>

<faultstring>Schema Validation Error(s)</faultstring>

<detail>

<Fault xmlns:xsd="http://socialservices.dss.gov.au/common">

<ReferenceID>xxxx-yyyy</ReferenceID>

<FaultCode>codexxx</FaultCode>

<Category>ValidationError</Category>

<FaultText>

<ValidationError>

<Element>PostCode</Element>

<Line>10</Line>

<Column>3</Column>

<Description>Invalid value according to its data type

</Description>

</ValidationError>

<ValidationError>

<Element>FamilyName</Element>

<Line>23</Line>

<Column>5</Column>

<Description>Element content is incomplete according to the DTD/Schema.

</Description>

</ValidationError>

</FaultText>

<RawFaultData />

<InstanceID>abac</InstanceID>

</Fault>

</detail>

</soap:Fault>

</soap:Body>

</soap:Envelope>

Within the **faultText** element, DSS will place a set of **ValidationError** elements, one for each error encountered. The **detail** element contains the following fields:

| SOAP Fault Element Name | Description |
| --- | --- |
| Element | The element in which the error occurred. |
| Line | The line number of the incoming request where the error occurred. |
| Column | The column number of the incoming request where the error occurred. |
| Description | A brief description of why the element was invalid. |

Line and column number will only be useful if the tools used to call web services allow access to view the raw SOAP request sent over the wire. In any case, the other fields (**Element** and **Description**) should be sufficient to pinpoint where the error occurred in the request.

#### System error example

An example of a server response to a request, which does not pass schema validation look like:

<?xml version="1.0" encoding="utf-8" standalone="yes" ?>

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<soap:Fault>

<faultcode>soap:Client</faultcode>

<faultstring>Schema Validation Error(s)</faultstring>

<detail>

<Fault xmlns:xsd="http://socialservices.dss.gov.au/common">

<ReferenceID>xxxx-yyyy</ReferenceID>

<FaultCode>codexxx</FaultCode>

<Category>ValidationError</Category>

<FaultText>

Access Denied

</FaultText>

<RawFaultData />

<InstanceID>abac</InstanceID>

</Fault>

</detail>

</soap:Fault>

</soap:Body>

</soap:Envelope>

#### Business errors

Once the incoming request has validated successfully against the schema, the server will process each transaction according to the respective business rules. Any errors or messages for a transaction data element arising from this validation are stored in the **TransactionStatus** field.

## Statistical Linkage Key (SLK)

Selected letters of **family** **name**, **given** **name**, **date** **of** **birth** and **gender** are used to generate a statistical linkage key to facilitate the linkage of records for statistical purposes only. The SLK ensures that individual clients will not be identified from data in the collection and comprises the alpha code derived from a client’s name, their sex, and full date of birth.

### SLK algorithm

The following algorithm is used to generate an SLK for use with the Data Exchange system.

* Record 2nd, 3rd and 5th letters of last name (family name).
* Record 2nd and 3rd letters of first (given) name.
* Record the date of birth as a character string of the form ddmmyyyy.
* Record the sex of the client as :-
  + Code 1 for Male
  + Code 2 for Female
  + Code 3 for Intersex or indeterminate
  + Code 9 for Not stated/inadequately described
* Do not count hyphens, apostrophes, blank spaces, or any other character that may appear in a name that is not a letter of the alphabet.
* Where the name is not long enough to supply all the requested letters, fill the remaining squares with a 2 to indicate that a letter does not exist. This will occur if the first name is less than 3 characters and if the last name is less than 5 characters. If a name is too short, complete with 2’s.
* Where a name or part of a name is missing, substitute a 9 to indicate that the letter is not known.
* Always use block capital letters for SLK.

### SLK regular expression

* The following regular expression may be useful for validation of your generated SLK.

Please note: The regular expression allows 29 days in any February, it does not check for leap years.

* ([9]{3}|[A-Z]([2]{2}|[A-Z][A-Z,2]))([9]{2}|[A-Z][A-Z,2])(((((0[1-9]|[1-2][0-9]))|(3[01]))((0[13578])|(1[02])))|((((0[1-9]|[1-2][0-9]))|(30))((0[469])|(11)))|((0[1-9]|[1-2][0-9])02))(19|2[0-9])[0-9]{2}[1 2 3 9]

### SLK examples

| First Name | Last Name | Gender | Date of Birth | SLK |
| --- | --- | --- | --- | --- |
| Jane | Citizen | Female | 27 May 1963 | ITZAN270519632 |
| Joseph | Bloggs | Male | 31 December 1959 | LOGOS311219591 |
| Jane | Luca | Female | 27 May 1963 | UC2AN270519632 |
| Jo | O’Donnell | Female | 27 May 1963 | DONO2270519632 |
| J | Bloggs | Female | 27 May 1963 | LOG99270519632 |
| J | Blog | Female | 27 May 1963 | LO299270519632 |
| J | O | Male | 31 December 1959 | 99999311219591 |
| J | Blog | Not stated | 20 June 1967 | LO299200619679 |
| Joseph | Bloggs | Intersex | 31 December 1959 | LOGOS311219593 |

# Interface method definitions

**The following sections detail the various methods within the interfaces provided by the Data Exchange System Web Service Interface.**

## DataCollection.Utilities

### Ping

The **Ping** **method** is a diagnostic method that can be used to test connectivity and authentication with the Data Exchange System Web service. This method provides no functionality other than to return an empty response. The intent of the interface is to assist in the testing and diagnosing connection and authentication issues.

#### Ping request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| Ping | Mandatory | N/A | N/A | An empty node to indicate a Ping Request. |

#### Ping example request message

<PingRequest>

</PingRequest>

#### Ping response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| VersionNo | String |  |  | The Data Exchange version number. |
| Environment | String |  |  | The Data Exchange environment. |

#### Ping example response message

<PingResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<VersionNo>1.0.0.0</VersionNo>

<Environment>Prod</Environment>

</PingResponse>

## DataCollection.Reference

### GetReferenceData

The **GetReferenceData** operation provides the reference data required to interact with the Data Exchange system database.

#### GetReferenceData request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| ReferenceDataCode | Mandatory | String | N/A | Valid values are:  All  AboriginalOrTorresStraitIslanderOrigin  AccommodationType  AgBusinessType  Ancestry  AssessedBy  AssessmentPhase  AttendanceProfile  Country  Disability  DVACardStatus  EducationLevel  EmploymentStatus  ExitReason  ExternalReferralDestination  ExtraItem  Gender  HardshipType  HomelessIndicator  HouseholdComposition  IncomeFrequency  Language  MainSourceOfIncome  MigrationVisaCategory  MoneyBusinessCommunityEducationWorkshop  NDISEligibility  ParentingAgreement  ParticipationType  PropertyAgreement  ReasonForAssistance  ReferralPurpose  ReferralSource  ReferralType  ScoreType  Section60ICertificateType  ServiceSetting  State  Topic |

#### GetReferenceData example request message

<GetReferenceDataRequest>

<ReferenceDataCode>Gender</ReferenceDataCode>

</GetReferenceDataRequest>

#### GetReferenceData response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| Data |  |  |  | Collection of ReferenceData values. |
| ReferenceData |  |  |  |  |
| Code | String |  |  | The ReferenceData value code. |
| Description | String |  |  | Description of ReferenceData value. |
| OrderNumber | Int |  |  | Default display order for reference item. |
| CodeType | String |  |  | Defining the reference data type that this reference data item belongs to. |
| UpdatedDateTime | DateTime |  |  | Last update date and time of reference data item. |
| MaxLastUpdatedDateTime | DateTime |  |  | Maximum Last updated date time of the result set. |

#### GetReferenceData example response message

<GetReferenceDataResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Data>

<ReferenceData>

<Code>MALE</Code>

<Description>Male</Description>

<OrderNumber>1</OrderNumber>

<CodeType>Gender</CodeType>

<UpdatedDateTime>2014-04-04T10:35:31.853</UpdatedDateTime>

</ReferenceData>

<ReferenceData>

<Code>FEMALE</Code>

<Description>Female</Description>

<OrderNumber>2</OrderNumber>

<CodeType>Gender</CodeType>

<UpdatedDateTime>2014-04-04T10:35:31.853</UpdatedDateTime>

</ReferenceData>

<ReferenceData >

<Code>INTERSEX</Code>

<Description>Intersex indeterminate</Description>

<OrderNumber>3</OrderNumber>

<CodeType>Gender</CodeType>

<UpdatedDateTime>2014-04-04T10:35:31.853</UpdatedDateTime>

</ReferenceData>

<ReferenceData>

<Code>NOTSTATED</Code>

<Description>Not stated/Inadequately described</Description>

<OrderNumber>4</OrderNumber>

<CodeType>Gender</CodeType>

<UpdatedDateTime>2014-04-04T10:35:31.853</UpdatedDateTime>

</ReferenceData>

</Data>

<MaxLastUpdatedDateTime>2014-04-04T10:35:31.853</ MaxLastUpdatedDateTime >

</GetReferenceDataResponse>

### GetAssessmentReferenceDetails

The **GetAssessmentReferenceDetails** operation provides the assessment reference data required to interact with the Data Exchange system database.

#### GetAssessmentReferenceData request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| ScoreTypeCode | Mandatory | One occurrence of enumeration. | N/A | Please use GetOrganisationActivities method to get available AvailableAssessmentTypes. |

#### GetAssessmentReferenceDetails example request message

<GetAssessmentReferenceDetailsRequest>

<!--type: string-->

<ScoreTypeCode>Group</ScoreTypeCode>

</GetAssessmentReferenceDetailsRequest>

#### GetAssessmentReferenceDetails response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| AssessmentReferenceDetail |  |  |  | Collection of ReferenceData values. |
| ScoreType | String |  |  | The AssessmentScoreType value code. |
| ApplicableFor | String |  |  | Session or Client. |
| Domains |  |  |  |  |
| Domain |  |  |  |  |
| DomainCode | String |  |  | The DomainCode value. |
| DomainDescription | String |  |  | Description of ScoreType value. |
| Scores |  |  |  |  |
| Score |  |  |  |  |
| ScoreCode | String |  |  | The Score code value. |
| Description | String |  |  | Description of Score value. |

#### GetAssessmentReferenceDetails example response message

<GetAssessmentReferenceDetailsResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<AssessmentReferenceDetail>

<ScoreType>GROUP</ScoreType>

<ApplicableFor>Session</ApplicableFor>

<Domains>

<Domain>

<DomainCode>GROUPNETWORKS</DomainCode>

<DomainDescription>Community infrastructure and networks</DomainDescription>

<Scores>

<Score>

<ScoreCode>GROUPNETWORKS1</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>GROUPNETWORKS2</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>GROUPNETWORKS3</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>GROUPNETWORKS4</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>GROUPNETWORKS5</ScoreCode>

<Description/>

</Score>

</Scores>

</Domain>

<Domain>

<DomainCode>GROUPSKILLS</DomainCode>

<DomainDescription>Group / community knowledge, skills, attitudes and behaviours</DomainDescription>

<Scores>

<Score>

<ScoreCode>GROUPSKILLS1</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>GROUPSKILLS2</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>GROUPSKILLS3</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>GROUPSKILLS4</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>GROUPSKILLS5</ScoreCode>

<Description/>

</Score>

</Scores>

</Domain>

<Domain>

<DomainCode>ORGSKILLS</DomainCode>

<DomainDescription>Organisational knowledge, skills and practices</DomainDescription>

<Scores>

<Score>

<ScoreCode>ORGSKILLS1</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>ORGSKILLS2</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>ORGSKILLS3</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>ORGSKILLS4</ScoreCode>

<Description/>

</Score>

<Score>

<ScoreCode>ORGSKILLS5</ScoreCode>

<Description/>

</Score>

</Scores>

</Domain>

</Domains>

</AssessmentReferenceDetail>

</GetAssessmentReferenceDetailsResponse>

## DataCollection.Organisation

### GetOrganisation

The GetOrganisation operation retrieves the Organisation details for the user executing the operation.

#### GetOrganisation request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| GetOrganisation | Mandatory | N/A | N/A |  | An empty node to indicate a GetOrganisation Request. |

#### GetOrganisation example request message

<GetOrganisationRequest>

</GetOrganisationRequest>

#### GetOrganisation response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| Organisation |  |  |  |  |
| Name | String |  |  | Organisation Name. |
| ABN | String |  |  | Organisation Australian Business Number. |
| ArrayOfOutlet | Array of Organisation Outlets |  |  |  |
| Outlet | Array of Outlets |  |  |  |
| OutletId | string |  |  |  |
| Name |  |  |  |  |
| OutletDetails |  |  |  |  |
| Address |  |  |  |  |
| Suburb | string |  |  |  |
| State | string |  |  |  |
| PostCode | string |  |  |  |
| AddressLine1 | string |  |  |  |
| AddressLine2 | string |  |  |  |
| OrganisationActivities | Array of Organisation Activities |  |  |  |
| OrganisationActivity |  |  |  |  |
| OrganisationActivityId | int |  |  |  |
| ProgramActivityName | string |  |  |  |
| StartDate | date |  |  |  |
| EndDate |  |  |  |  |
| DeliveringOnBehalfOfAnotherOrganisation | Boolean |  |  |  |
| HasPartnershipEnded | Boolean |  |  |  |
| DeliveryPartners | Array of Delivery Partner |  |  |  |
| DeliveryPartner |  |  |  |  |
| PartnerOrganisationName | string |  |  |  |
| PartnerOrganisationExternalSystemId | string |  |  |  |
| PartnerOrganisationABN | string |  |  |  |

#### GetOrganisation example response message

<GetOrganisationResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Organisation>

<Name>Organisation Name</Name>

<ABN>57370813702</ABN>

<ArrayOfOutlet>

<Outlet>

<OutletId>8</OutletId>

<Name>AAA Australian Support Services</Name>

<OutletDetails>

<Address>

<Suburb>GLENELG NORTH</Suburb>

<State>SA</State>

<PostCode>5045</PostCode>

<AddressLine1/>

<AddressLine2>10 Darwin Street</AddressLine2>

</Address>

</OutletDetails>

</Outlet>

</ArrayOfOutlet>

<OrganisationActivities>

<OrganisationActivity>

<OrganisationActivityId>59</OrganisationActivityId>

<ProgramActivityName>Children's Contact Services (for AAA Help Mod)</ProgramActivityName>

<StartDate>2012-01-01T00:00:00</StartDate>

<EndDate>2015-12-31T00:00:00</EndDate>

<DeliveringOnBehalfOfAnotherOrganisation>true</DeliveringOnBehalfOfAnotherOrganisation>

<HasPartnershipEnded>false</HasPartnershipEnded>

<DeliveryPartners>

<DeliveryPartner>

<PartnerOrganisationName>AAA Help Mod</PartnerOrganisationName>

<PartnerOrganisationExternalSystemId/>

<PartnerOrganisationABN>48123123124</PartnerOrganisationABN>

</DeliveryPartner>

</DeliveryPartners>

</OrganisationActivity>

</OrganisationActivities>

</Organisation>

</GetOrganisationResponse>

### GetOrganisationActivities

The **GetOrganisationActivities** operation retrieves the organisation activity details for the user executing the operation.

#### GetOrganisationActivities request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| GetOrganisationActivities | Mandatory | N/A | N/A |  | An empty node to indicate a GetOrganisationActivities Request. |

#### GetOrganisationActivities example request message

<GetOrganisationActivitiesRequest>

</GetOrganisationActivitiesRequest>

#### GetOrganisationActivities response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| OrganisationActivities | Array of Organisation Activities |  |  |  |
| OrganisationActivity |  |  |  |  |
| OrganisationActivityId | int |  |  |  |
| ProgramActivityName | string |  |  |  |
| StartDate | date |  |  | OrganisationActivity start date. |
| EndDate | date |  |  | OrganisationActivity end date. |
| DeliveringOnBehalfOfAnotherOrganisation | Boolean |  |  |  |
| HasPartnershipEnded | Boolean |  |  |  |
| DeliveryPartners | Array of Delivery partner |  |  |  |
| DeliveryPartner |  |  |  |  |
| PartnerOrganisationName | string |  |  |  |
| PartnerOrganisationExternalSystemId | string |  |  |  |
| PartnerOrganisationABN | string |  |  |  |
| ActivityServiceTypes |  |  |  |  |
| ServiceType |  |  |  |  |
| ServiceTypeId | int |  |  | Service Type Id to be used with session. |
| Name | string |  |  | Service Type Name. |
| ServiceTypeGroupName | string |  |  |  |
| ServiceTypeSpecificRequirements |  |  |  |  |
| ServiceTypeSpecificRequirement |  |  |  |  |
| Code | string |  |  |  |
| Description | string |  |  |  |
| AffiliateWith | string |  |  |  |
| AvailableFrom | date |  |  | Start date of Service Type Specific Requirement availability for sessions. |
| AvailableTo | date |  |  | End date of Service Type Specific Requirement availability for sessions.  Note: If not present or null, then end is not specified and Service Type Specific Requirement is assumed to be applicable from the specified start date. |
| ActivitySpecificRequirements |  |  |  |  |
| ActivitySpecificRequirement |  |  |  |  |
| Code | string |  |  |  |
| Description | string |  |  |  |
| AffiliateWith | string |  |  |  |
| AvailableFrom | date |  |  | Start date of Activity Specific Requirement availability for sessions. |
| AvailableTo | date |  |  | End date of Activity Specific Requirement availability for sessions.  Note: if not present or null, then end is not specified and Activity Specific Requirement is assumed to be applicable from the specified start date. |
| AvailableAssessmentTypes |  |  |  |  |
| AvailableAssessmentType |  |  |  |  |
| AssessmentTypeCode | string |  |  |  |
| Description | string |  |  |  |
| AvailableForEntity | string |  |  |  |
| AvailableFrom | date |  |  | Start date of Assessment Type availability for sessions. |
| AvailableTo | date |  |  | End date of Assessment Type availability for sessions.  Note: if not present or null, then end is not specified and Activity Specific Requirement is assumed to be applicable from the specified start date. |

#### GetOrganisationActivities example response message

<GetOrganisationActivitiesResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<OrganisationActivities>

<OrganisationActivity>

<OrganisationActivityId>59</OrganisationActivityId>

<ProgramActivityName>Children's Contact Services (for AAA Help Mod)</ProgramActivityName>

<StartDate>2012-01-01T00:00:00</StartDate>

<EndDate>2015-12-31T00:00:00</EndDate>

<DeliveringOnBehalfOfAnotherOrganisation>true</DeliveringOnBehalfOfAnotherOrganisation>

<HasPartnershipEnded>false</HasPartnershipEnded>

<DeliveryPartners>

<DeliveryPartner>

<PartnerOrganisationName>AAA Help Mod</PartnerOrganisationName>

<PartnerOrganisationExternalSystemId/>

<PartnerOrganisationABN>48123123124</PartnerOrganisationABN>

</DeliveryPartner>

</DeliveryPartners>

<ActivityServiceTypes>

<ServiceType>

<ServiceTypeId>39</ServiceTypeId>

<Name>Psychologist</Name>

<ServiceTypeGroupName>Allied Health and Therapy Services</ServiceTypeGroupName>

<ServiceTypeSpecificRequirements>

<ServiceTypeSpecificRequirement>

<Code>ASSISTANCEMINUTES</Code>

<Description>When applied to a service type “Time” property on session is required and must be supplied with a valid value to record a new session or update an existing session.</Description>

<AffiliateWith>Session</AffiliateWith>

<AvailableFrom>2014-07-01T00:00:00</AvailableFrom>

</ServiceTypeSpecificRequirement>

</ServiceTypeSpecificRequirements>

<AvailableFrom>2014-07-01T00:00:00</AvailableFrom>

</ServiceType>

<ActivityServiceTypes>

<ActivitySpecificRequirements>

<ActivitySpecificRequirement>

<Code>TOPIC</Code>

<Description>When applied to an Activity, “Topic” property on Session must be supplied.</Description>

<AffiliateWith>Session</AffiliateWith>

<AvailableFrom>2014-07-01T00:00:00</AvailableFrom>

</ActivitySpecificRequirement>

</ActivitySpecificRequirements>

<AvailableAssessmentTypes>

<AvailableAssessmentType>

<AssessmentTypeCode>CIRCUMSTANCES</AssessmentTypeCode>

<Description>Circumstances</Description>

<AvailableForEntity>Client</AvailableForEntity>

<AvailableFrom>2014-07-01T00:00:00</AvailableFrom>

</AvailableAssessmentType>

</AvailableAssessmentTypes>

</OrganisationActivity>

</OrganisationActivities>

</GetOrganisationActivitiesResponse>

### GetOutlet

The **GetOutlet** operation retrieves the outlet details registered with the Data Exchange.

#### GetOutlet request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| OutletId | Mandatory | Int | N/A |  | DEX’s OutletId. The user should have access to the Outlet. |

#### GetOutlet example request message

<GetOutletRequest>

<!--type int-->

<OutletId>606</OutletId>

</GetOutletRequest>

#### GetOutlet response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| Outlet |  |  |  |  |
| OutletId | Int |  |  | DEX’s OutletId. |
| Name | String |  |  | DEX’s Outlet Name. |
| OutletDetails |  |  |  |  |
| Address |  |  |  | Outlet Location. |
| Suburb | String |  |  |  |
| State | String |  |  |  |
| Postcode | String |  |  |  |
| AddressLine1 | String |  |  |  |
| AddressLine2 | String |  |  |  |

#### GetOutlet example response message

<GetOutletResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Outlet>

<OutletId>606</OutletId>

<Name>Kalparrin Community Inc</Name>

<OutletDetails>

<Address>

<Suburb>MURRAY BRIDGE</Suburb>

<State>SA</State>

<Postcode>5253</Postcode>

<AddressLine1>1 Main Street</AddressLine1>

<AddressLine2/>

</Address>

</OutletDetails>

</Outlet>

</GetOutletResponse>

### GetOutletActivities

The **GetOutletActivities** operation retrieves all the active **OutletActivities** registered with the Data Exchange to which user has access.

#### GetOutletActivities request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| GetOutletActivities | Mandatory | N/A | N/A |  | An empty node to indicate a GetOutletActivities Request. |

#### GetOutletActivities example request message

<GetOutletActivitiesRequest>

</GetOutletActivitiesRequest>

#### GetOutletActivities response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| OutletActivities | Array (OutletActivity) |  |  |  |
| OutletActivity |  |  |  |  |
| OutletActivityId | Int |  |  | DEX’s OutletActivityId. |
| OutletId | Int |  |  | DEX’s OutletId. |
| OutletName | String |  |  | DEX’s Outlet Name. |
| ActivityName | String |  |  | DEX’s Activity Name. |
| ProgramName | String |  |  | DEX’s Program name. |
| OrganisationActivityId | Int |  |  | Organisation Activity Id. |
| StartDate | Date |  |  | StartDate of OutletActivity. |
| EndDate | Date |  |  | EndDate of OutletActivity. |
| Status | String |  |  |  |

#### GetOutletActivities example response message

<GetOutletActivitiesResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<OutletActivities>

<OutletActivity>

<OutletActivityId>219</OutletActivityId>

<OutletId>171</OutletId>

<OutletName>Outlet Name</OutletName>

<ActivityName>Activity Name</ActivityName>

<ProgramName>Family Support</ProgramName>

<OrganisationActivityId>1194</OrganisationActivityId>

<StartDate>2009-07-01T000000</StartDate>

<EndDate>2014-06-10T000000</EndDate>

<Status>APPROVED</Status>

</OutletActivity>

<OutletActivity>

<OutletActivityId>993</OutletActivityId>

<OutletId>171</OutletId>

<OutletName>Outlet Name</OutletName>

<ActivityName>Activity Name</ActivityName>

<ProgramName>Family Support</ProgramName>

<OrganisationActivityId>1195</OrganisationActivityId>

<StartDate>2011-07-01T000000</StartDate>

<EndDate>2014-06-30T000000</EndDate>

<Status>APPROVED</Status>

</OutletActivity>

</OutletActivities>

</GetOutletActivitiesResponse>

### GetUser

The GetUser operation retrieves the User details for the user executing the operation.

#### GetUser request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| GetUser | Mandatory | N/A | N/A |  | An empty node to indicate a GetUser Request. |

#### GetUser example request message

<GetUserRequest>

</GetUserRequest>

#### GetUser response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| User |  |  |  |  |
| UserName | String |  |  | User Name. |
| FirstName | String |  |  | Frist Name of the user. |
| FamilyName | String |  |  | Family Name of the user. |
| Phone | String |  |  | Phone number of the user. |
| Email | String |  |  | Email address of the user. |
| Organisation |  |  |  |  |
| Name | String |  |  | Organisation Name. |
| ABN | String |  |  | Organisation Australian Business Number. |
| OutletActivities | Array (OutletActivity) |  |  |  |
| OutletActivityId | Int |  |  | DEX’s Outlet Activity Identifier. |
| OutletId | Int |  |  | DEX’s Outlet Identifier. |
| OutletName | String |  |  | DEX’s Outlet Name. |
| ActivityName | String |  |  | Activity Name. |
| ProgramName | String |  |  | Program Name. |
| OrganisationActivityId | Int |  |  | Organisation activity Id. |
| StartDate | Date |  |  | StartDate of OutletActivity. |
| EndDate | Date |  |  | EndDate of OutletActivity. |
| Status | string |  |  |  |

#### GetUser example response message

<GetUserResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<User>

<UserName>jb0012@production.local</UserName>

<FirstName>Jo</FirstName>

<FamilyName>Blog</FamilyName>

<Phone/>

<Email/>

<Organisation>

<Name>Warmun Community (Turkey Creek) Inc</Name>

<ABN>57370813702</ABN>

</Organisation>

<OutletActivities>

<OutletActivity>

<OutletActivityId>219</OutletActivityId>

<OutletId>171</OutletId>

<OutletName>Warmun Early Learning Centre</OutletName>

<ActivityName>Family Law Counselling</ActivityName>

<ProgramName>Family Support (Bill 1)</ProgramName>

<OrganisationActivityId>1195</OrganisationActivityId>

<StartDate>2009-07-01T00:00:00</StartDate>

<EndDate>2014-06-10T00:00:00</EndDate>

<Status>APPROVED</Status>

</OutletActivity>

<OutletActivity>

<OutletActivityId>993</OutletActivityId>

<OutletId>171</OutletId>

<OutletName>Warmun Early Learning Centre</OutletName>

<ActivityName>Financial capability</ActivityName>

<ProgramName>Family Support (Bill 1)</ProgramName>

<OrganisationActivityId>1195</OrganisationActivityId>

<StartDate>2011-07-01T00:00:00</StartDate>

<EndDate>2014-06-30T00:00:00</EndDate>

<Status>APPROVED</Status>

</OutletActivity>

</OutletActivities>

</User>

</GetUserResponse>

## DataCollection.Recipient

### AddClient

The **AddClient** operation is to be used when an organisation wishes to submit data to the Data Exchange in order to create and register a Client with their organisation. During the creation process, validation of the data submitted by the organisation will be undertaken. If the validation of the data is unsuccessful, an error will be generated and no record will be created.

A client should only be registered once with an organisation. A registered client can be attached to many client cases.

IMPORTANT: The system does not identify ‘pseudonym’ as a field. All instances of ‘pseudonym’ must be spelled as ‘psuedonym’. This is due to an error in the initial system build.

#### AddClient request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| ClientId | Mandatory | String | 30 |  | Organisation’s Client Id.  It has to be unique for all the clients registered with the Organisation. It is **not** case sensitive.  A client’s name, part of a client’s name, or other identifiable information should not be used as a ClientId under any circumstances. |
| SLK | Conditional  Mandatory | String | 14 |  | Must be supplied if ConsentToProvideDetails is false. |
| ConsentToProvideDetails | Mandatory | Boolean |  |  | Client consent for DSS to collect personal information from organisations for storage on the Data Exchange. |
| ConsentedForFutureContacts | Mandatory | Boolean |  |  | Client consent to future contact for survey / research / evaluation. |
| GivenName | Conditional  Mandatory | String | 30 |  | Must be supplied if ConsentToProvideDetails is true. |
| FamilyName | Conditional  Mandatory | String | 30 |  | Must be supplied if ConsentToProvideDetails is true. |
| IsUsingPsuedonym | Mandatory | Boolean |  |  | true = If name specified is a Psuedonym name  false = If name specified is not a Psuedonym name. |
| BirthDate | Mandatory | DateTime |  |  | If IsBirthDateAnEstimate is true then it should be in the format of ‘yyyy-01-01’.  If IsBirthDateAnEstimate is false then it should be the real birth date in the format of ‘yyyy-mm-dd’. |
| IsBirthDateAnEstimate | Mandatory | Boolean |  |  | true = the birth date is just an estimate  false = the birth date is the real birth date. |
| GenderCode | Mandatory | String |  |  | Must be a valid Gender code retrieved using GetReferenceData operation. |
| ResidentialAddress |  |  |  |  |  |
| Suburb | Mandatory | String | 50 |  | Suburb from residential address of a client. |
| State | Mandatory | String | 3 |  | State from residential address of a client. |
| Postcode | Mandatory | String | 4 |  | Postcode from residential address of a client. |
| AddressLine1 | Optional | String | 180 |  | Street address from residential address of a client. |
| AddressLine2 | Optional | String | 180 |  | Overflow from Address line 1 from residential address of a client. |
| CountryOfBirthCode | Mandatory | String |  |  | Must be a valid Country code retrieved using GetReferenceData operation. |
| LanguageSpokenAtHomeCode | Mandatory | String |  |  | Must be a valid Language code retrieved using GetReferenceData operation. |
| AboriginalOrTorresStraitIslanderOriginCode | Mandatory | String |  |  | Must be a valid AboriginalOrTorresStraitIslanderOrigin code retrieved using GetReferenceData operation. |
| HasDisabilities | Mandatory | Boolean |  |  | If True, then minimum of one Disability code is required.  If False, then no Disability code should be sent. |
| HomelessIndicatorCode | Optional | String |  |  | Must be a valid HomelessIndicatorCode retrieved using GetReferenceData operation. |
| HouseholdCompositionCode | Conditional  Optional | String |  |  | Must be a valid HouseholdCompositionCode retrieved using GetReferenceData operation.  Also named as ‘Living arrangements’. |
| MainSourceOfIncomeCode | Optional | String |  |  | Must be a valid MainSourceOfIncomeCode retrieved using GetReferenceData operation. |
| IncomeFrequencyCode | Optional | String |  |  | Must be a valid IncomeFrequency code retrieved using GetReferenceData operation. |
| IncomeAmount | Optional | Int |  |  | Client income. |
| FirstArrivalMonth | Optional | String |  |  | Month Client first arrived in Australia  Full name of the month:  January, February, March, April, May, June, July, August, September, October, November, December |
| FirstArrivalYear | Optional | Int |  |  | Year Client first arrived in Australia. |
| MigrationVisaCategoryCode | Optional | String |  |  | Must be a valid MigrationVisaCategoryCode retrieved using GetReferenceData operation. |
| AncestryCode | Optional | String |  |  | Must be a valid AncestryCode retrieved using GetReferenceData operation. |
| AccommodationTypeCode | Conditional  Optional | String |  |  | Must be a valid AccommodationType retrieved using GetReferenceData operation. |
| DVACardStatusCode | Conditional  Optional | String |  |  | Must be a valid DVACardStatus retrieved using GetReferenceData operation. |
| HasCarer | Conditional  Optional | Boolean |  |  | true = If client has a carer  false = If client does not have a carer |
| EducationLevelCode | Optional | String |  |  | Must be a valid EducationLevelCode retrieved using GetReferenceData operation. |
| EmploymentStatusCode | Optional | String |  |  | Must be a valid EmploymentStatusCode retrieved using GetReferenceData operation. |
| IsClientACarer | Optional | Boolean |  |  | true = If client is a carer  false = If client is not a carer |
| NDISEligibilityCode | Optional | String |  |  | Must be a valid NDISEligibilityCode retrieved using GetReferenceData operation. |
| Disabilities | Optional |  |  |  |  |
| DisabilityCode | Mandatory | String |  |  | Must be a valid DisabilityCode retrieved using GetReferenceData operation. |
| HasValidatedForDuplicateClient |  | Boolean |  |  |  |

#### AddClient example request message

<AddClientRequest>

<Client>

<!--type: string-->

<ClientId>CL0012</ClientId>

<!--type: string-->

<SLK>LO222211220001</SLK>

<!--type: Boolean-->

<ConsentToProvideDetails>true</ConsentToProvideDetails>

<!--type: Boolean-->

<ConsentedForFutureContacts>true</ConsentedForFutureContacts>

<!--type: string-->

<GivenName>Joe</GivenName>

<!--type: string-->

<FamilyName>Blog</FamilyName>

<!--type: Boolean-->

<IsUsingPsuedonym>true</IsUsingPsuedonym>

<!--type: dateTime-->

<BirthDate>2000-12-21T00:00:00</BirthDate>

<!--type: Boolean-->

<IsBirthDateAnEstimate>false</IsBirthDateAnEstimate>

<!--type: string-->

<GenderCode>Male</GenderCode>

<ResidentialAddress>

<!--type: string-->

<Suburb>Calwell</Suburb>

<!--type: string-->

<State>ACT</State>

<!--type: string-->

<Postcode>2905</Postcode>

<!--Optional:-->

<!--type: string-->

<AddressLine1>1 temp pl</AddressLine1>

<!--Optional:-->

<!--type: string-->

<AddressLine2/>

</ResidentialAddress>

<!--type: string-->

<CountryOfBirthCode>1101</CountryOfBirthCode>

<!--type: string-->

<LanguageSpokenAtHomeCode>1201</LanguageSpokenAtHomeCode>

<!--type: string-->

<AboriginalOrTorresStraitIslanderOriginCode>NOTSTATED</AboriginalOrTorresStraitIslanderOriginCode>

<!--type: Boolean-->

<HasDisabilities>true</HasDisabilities>

<!--Optional:-->

<!--type: string-->

<HomelessIndicatorCode>ATRISK</HomelessIndicatorCode>

<!--Optional:-->

<!--type: string-->

<HouseholdCompositionCode>SINGLE</HouseholdCompositionCode>

<!--Optional:-->

<!--type: string-->

<MainSourceOfIncomeCode>EMPLOYED</MainSourceOfIncomeCode>

<!--Optional:-->

<!--type: string-->

<IncomeFrequencyCode>WEEKLY</IncomeFrequencyCode>

<!--Optional:-->

<!--type: int-->

<IncomeAmount>300</IncomeAmount>

<!--Optional:-->

<!--type: Month - enumeration: [January,February,March,April,May,June,July,August,September,October,November,December]-->

<FirstArrivalMonth>January</FirstArrivalMonth>

<!--Optional:-->

<!--type: int-->

<FirstArrivalYear>1999</FirstArrivalYear>

<!--Optional:-->

<!--type: string-->

<MigrationVisaCategoryCode>HUMANITARIAN</MigrationVisaCategoryCode>

<!--Optional:-->

<!--type: string-->

<AncestryCode>1103</AncestryCode>

<!--Optional:-->

<!--type: string-->

<AccommodationTypeCode>25</AccommodationTypeCode>

<!--Optional:-->

<!--type: string-->

<DVACardStatusCode>GOLD</DVACardStatusCode>

<!--Optional:-->

<!--type: Boolean-->

<HasCarer>false</HasCarer>

<!--Optional:-->

<!--type: string-->

<EducationLevelCode>PREPRIMARY</EducationLevelCode>

<!--Optional:-->

<!--type: string-->

<EmploymentStatusCode>WORKFULLTIME</EmploymentStatusCode>

<!--Optional:-->

<!--type: Boolean-->

<IsClientACarer>false</IsClientACarer>

<!--Optional:-->

<!--type: string-->

<NDISEligibilityCode>ELIGIBLE</NDISEligibilityCode>

<Disabilities>

<!--Zero or more repetitions:-->

<!--type: string-->

<dex:DisabilityCode>learning</dex:DisabilityCode>

</Disabilities>

</Client>

<!--type: Boolean-->

<HasValidatedForDuplicateClient>true</HasValidatedForDuplicateClient>

</AddClientRequest>

#### AddClient response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### AddClient example response message

<AddClientResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</AddClientResponse>

### GetClient

Retrieves Data Exchange details for Client registered for the user’s Organisation.

#### GetClient request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| ClientId | Mandatory | string | 30 |  | Organisation’s Client Id. |

#### GetClient example request message

<GetClientRequest>

<!--type: string-->

<ClientId>CL0012</ClientId>

</GetClientRequest>

#### GetClient response message definition

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |
| Client |  |  |  |
| ClientId | String |  | Organisation’s Client Id. |
| SLK | String |  | Supplied Or Generated SLK. |
| ConsentToProvideDetails | Boolean |  | Client consent for DSS to collect personal information from organisations for storage on the Data Exchange. |
| ConsentedForFutureContacts | Boolean |  | Client consent to future contact for survey / research / evaluation. |
| GivenName | String |  | Given name for client. |
| FamilyName | String |  | Family name for client. |
| IsUsingPsuedonym | Boolean |  | Specifying whether the name is a psuedonym name. |
| BirthDate | DateTime |  | Client’s date of birth. |
| IsBirthDateAnEstimate | Boolean |  | Specifying whether the date of birth is an Estimate. |
| GenderCode | String |  | Client’s Gender code. |
| ResidentialAddress |  |  |  |
| Suburb | String |  | Suburb from residential address of a client. |
| State | String |  | State from residential address of a client. |
| Postcode | String |  | Postcode from residential address of a client. |
| AddressLine1 | String |  | Street address from residential address of a client. |
| AddressLine2 | String |  | Overflow from Address line 1 from residential address of a client. |
| CountryOfBirthCode | String |  | The CountryOfBirth code value. |
| LanguageSpokenAtHomeCode | String |  | The LanguageSpokenAtHome code value. |
| AboriginalOrTorresStraitIslanderOriginCode | String |  | The AboriginalOrTorresStraitIslanderOrigin code value. |
| HasDisabilities | Boolean |  | If True, then minimum of one Disability code is required.  If False, then no Disability code should be sent. |
| HomelessIndicatorCode | String |  | The HomelessIndicator code value. |
| HouseholdCompositionCode | String |  | The HouseholdComposition code value. |
| MainSourceOfIncomeCode | String |  | The MainSourceOfIncome code value. |
| IncomeFrequencyCode | String |  | The IncomeFrequency code value. |
| IncomeAmount | Int |  | Client income. |
| FirstArrivalMonth | String |  | Month Client first arrived in Australia.  Full name of the month: January, February, March, April, May, June, July, August, September, October, November, December |
| FirstArrivalYear | Int |  | Year Client first arrived in Australia. |
| MigrationVisaCategoryCode | String |  | The MigrationVisaCategoryCode value. |
| AncestryCode | String |  | The AncestryCode value. |
| AccommodationTypeCode | String |  | The AccommodationType value. |
| DVACardStausCode | String |  | The DVACardStatus value. |
| HasCarer | Boolean |  | true = If client has a carer  false = If client does not have a carer |
| EducationLevelCode | String |  | The EducationLevelCode value. |
| EmploymentStatusCode | String |  | The EmploymentStatusCode value. |
| IsClientACarer | Boolean |  | true = If client is a carer  false = If client is not a carer |
| NDISEligibilityCode | String |  | The NDISEligibilityCode value. |
| Disabilities | Array |  |  |
| DisabilityCode | String |  | The DisabilityCode value. |
| CreatedDateTime | DateTime |  | DateTime Client was created in the Data Exchange’s database. |
| Cases | Array |  |  |
| CaseId | String |  | Cases on which the ClientId is attached will be retrieved. |

#### GetClient example response message

<GetClientResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Client>

<ClientId>CL0012</ClientId>

<SLK>LO2OE240220141</SLK>

<ConsentToProvideDetails>true</ConsentToProvideDetails>

<ConsentedForFutureContacts>true</ConsentedForFutureContacts>

<ConsentGivenToDepartment>true</ConsentGivenToDepartment>

<GivenName>Joe</GivenName>

<FamilyName>Blog</FamilyName>

<IsUsingPsuedonym>false</IsUsingPsuedonym>

<BirthDate>2014-02-24T00:00:00</BirthDate>

<IsBirthDateAnEstimate>false</IsBirthDateAnEstimate>

<GenderCode>MALE</GenderCode>

<ResidentialAddress>

<Suburb>Forde</Suburb>

<State>ACT</State>

<Postcode>2914</Postcode>

<AddressLine1/>

<AddressLine2/>

</ResidentialAddress>

<CountryOfBirthCode>1101</CountryOfBirthCode>

<LanguageSpokenAtHomeCode>1201</LanguageSpokenAtHomeCode>

<AboriginalOrTorresStraitIslanderOriginCode>NO</AboriginalOrTorresStraitIslanderOriginCode>

<HasDisabilities>true</HasDisabilities>

<HomelessIndicatorCode>ATRISK</HomelessIndicatorCode>

<HouseholdCompositionCode>SINGLE</HouseholdCompositionCode>

<MainSourceOfIncomeCode>NIL</MainSourceOfIncomeCode>

<IncomeFrequencyCode>WEEKLY</IncomeFrequencyCode>

<IncomeAmount>300</IncomeAmount>

<FirstArrivalMonth>January</FirstArrivalMonth>

<FirstArrivalYear>2002</FirstArrivalYear>

<MigrationVisaCategoryCode>HUMANITARIAN</MigrationVisaCategoryCode>

<AncestryCode>2102</AncestryCode>

<AccommodationTypeCode>1</AccommodationTypeCode>

<DVACardStatusCode>1</DVACardStatusCode>

<HasCarer>false</HasCarer>

<EducationLevelCode>PREPRIMARY</EducationLevelCode>

<EmploymentStatusCode>WORKFULLTIME</EmploymentStatusCode>

<IsClientACarer>false</IsClientACarer>

<NDISEligibilityCode>ELIGIBLE</NDISEligibilityCode>

<Disabilities>

<DisabilityCode>PHYSICAL</<DisabilityCode>

</Disabilities>

<CreatedDateTime>2015-06-17T14:47:07.407</CreatedDateTime>

<Cases>

<CaseId>0054</CaseId>

<CaseId>0007</CaseId>

</Cases>

</Client>

</GetClientResponse>

### SearchClient

**SearchClient** operation can be used to search for the clients registered for the user’s organisation based on the specified criteria.

#### SearchClient request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| Search Fields (  Search Request Message Definition) |  |  |  |  |  |
| GivenName | Optional | String |  |  |  |
| FamilyName | Optional | String |  |  |  |
| ClientId | Optional | String |  |  | Organisation’s Client Id. |
| CreatedDateFrom | Optional | DateTime |  |  |  |
| CreatedDateTo | Optional | DateTime |  |  |  |
| Tags | Optional | Array |  |  |  |
| Tag | Optional | String |  |  | Only Clients that have a matching Tag or Tag(s) will be returned. If multiple Tag elements are supplied, then the Client must have all Tags to be returned.  Current available DEX tag : MissingRequiredData |

#### SearchClient example request message

<SearchClientRequest>

<Criteria>

<!--type: int-->

<PageIndex>1</PageIndex>

<!--type: int-->

<PageSize>100</PageSize>

<!--type: Boolean-->

<IsAscending>false</IsAscending>

<!--type: ClientSearchSortColumns - enumeration: [CreatedDate,ClientId,Name,BirthDate,Gender]-->

<SortColumn>CreatedDate</SortColumn>

<!--Optional:-->

<!--type: string-->

<GivenName>Joe</GivenName>

<!--Optional:-->

<!--type: string-->

<FamilyName>Blog</FamilyName>

<!--Optional:-->

<!--type: string-->

<ClientId>CL0012</ClientId>

<!--Optional:-->

<!--type: dateTime-->

<CreatedDateFrom>2014-09-20T16:10:00.000</CreatedDateFrom>

<!—Optional:-->

<!—type: dateTime-->

<CreatedDateTo>2014-09-25T00:00:00.000</CreatedDateTo>

<!—Optional:-->

<Tags>

<!—Zero or more repetitions:-->

<!—type: string-->

<Tag>MissingRequiredData</Tag>

</Tags>

</Criteria>

</SearchClientRequest>

#### SearchClient response message definition

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| Clients | Array (Client) |  |  | Collection of clients that meet the search criteria. |
| Client (  Client Response Message Definition) |  |  |  |  |
| TotalCount | Int |  |  | Number of matching records found. |

#### SearchClient example response message

<SearchClientResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Clients>

<Client>

<ClientId>CL0012</ClientId>

<SLK>LO2OE240220141</SLK>

<ConsentToProvideDetails>true</ConsentToProvideDetails>

<ConsentedForFutureContacts>true</ConsentedForFutureContacts>

<ConsentGivenToDepartment>true</ConsentGivenToDepartment>

<GivenName>Joe</GivenName>

<FamilyName>Blog</FamilyName>

<IsUsingPsuedonym>false</IsUsingPsuedonym>

<BirthDate>2014-02-24T00:00:00</BirthDate>

<IsBirthDateAnEstimate>false</IsBirthDateAnEstimate>

<GenderCode>MALE</GenderCode>

<ResidentialAddress>

<Suburb>Forde</Suburb>

<State>ACT</State>

<Postcode>2914</Postcode>

<AddressLine1/>

<AddressLine2/>

</ResidentialAddress>

<CountryOfBirthCode>1101</CountryOfBirthCode>

<LanguageSpokenAtHomeCode>1201</LanguageSpokenAtHomeCode>

<AboriginalOrTorresStraitIslanderOriginCode>NO</AboriginalOrTorresStraitIslanderOriginCode>

<HasDisabilities>true</HasDisabilities>

<HomelessIndicatorCode>ATRISK</HomelessIndicatorCode>

<HouseholdCompositionCode>GROUP</HouseholdCompositionCode>

<MainSourceOfIncomeCode>NIL</MainSourceOfIncomeCode>

<IncomeFrequencyCode>FORTNIGHTLY</IncomeFrequencyCode>

<IncomeAmount>1</IncomeAmount>

<FirstArrivalYear>2014</FirstArrivalYear>

<MigrationVisaCategoryCode>SKILLED</MigrationVisaCategoryCode>

<AncestryCode>2102</AncestryCode>

<AccommodationTypeCode>1</AccommodationTypeCode>

<DVACardStatusCode>1</DVACardStatusCode>

<HasCarer>false</HasCarer>

<EducationLevelCode>PREPRIMARY</EducationLevelCode>

<EmploymentStatusCode>WORKFULLTIME</EmploymentStatusCode>

<IsClientACarer>false</IsClientACarer>

<NDISEligibilityCode>ELIGIBLE</NDISEligibilityCode>

<Disabilities>

<DisabilityCode>Physical</DisabilityCode>

</Disabilities>

<CreatedDateTime>2014-09-22T16:19:09.317</CreatedDateTime>

</Client>

</Clients>

<TotalCount>1</TotalCount>

</SearchClientResponse>

### ValidateForDuplicateClient

**ValidateForDuplicateClient** provides a list of clients registered for the user’s organisation based on the supplied criteria.

#### ValidateForDuplicateClient request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| Search Fields (  Search Request Message Definition) |  |  |  |  |  |
| GivenName | Mandatory | String |  |  | Clients matching any three from the GivenName, FamilyName, BirthDate and Gender are retrieved. |
| FamilyName | Mandatory | String |  |  | Clients matching any three from the GivenName, FamilyName, BirthDate and Gender are retrieved. |
| ClientId | Mandatory | String |  |  | Organisation’s Client Id.  Any Client matching the ClientId will be retrieved. |
| GenderCode | Mandatory | String |  |  | Clients matching any three from the GivenName, FamilyName, BirthDate and Gender are retrieved. |
| BirthDate | Mandatory | Date |  |  | Clients matching any three from the GivenName, FamilyName, BirthDate and Gender are retrieved. |

#### ValidateForDuplicateClient example request message

<ValidateForDuplicateClientRequest>

<Criteria>

<!--type: int-->

<PageIndex>1</PageIndex>

<!--type: int-->

<PageSize>10</PageSize>

<!--type: Boolean-->

<IsAscending>true</IsAscending>

<!--type: ClientSearchSortColumns - enumeration: [CreatedDate,ClientId,Name,BirthDate,Gender]-->

<SortColumn>CreatedDate</SortColumn>

<!--type: string-->

<GivenName>Joe</GivenName>

<!--type: string-->

<FamilyName>Blog</FamilyName>

<!--type: string-->

<ClientId>CL0012</ClientId>

<!--type: string-->

<GenderCode>Male</GenderCode>

<!--type: dateTime-->

<BirthDate>2014-02-24T00:00:00</BirthDate>

</Criteria>

</ValidateForDuplicateClientRequest>

#### ValidateForDuplicateClient response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |
| Clients | Array (Client) |  |  | Collection of clients that meet the search criteria. |
| Client (  Client Response Message Definition) |  |  |  |  |
| TotalCount | Int |  |  | Number of matching records found. |

#### ValidateForDuplicateClient example response message

<ValidateForDuplicateClientResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Clients>

<Client>

<ClientId>CLB00016</ClientId>

<SLK>LO2OE300619851</SLK>

<ConsentToProvideDetails>true</ConsentToProvideDetails>

<ConsentedForFutureContacts>true</ConsentedForFutureContacts>

<ConsentGivenToDepartment>true</ConsentGivenToDepartment>

<GivenName>Joe</GivenName>

<FamilyName>Blog</FamilyName>

<IsUsingPsuedonym>false</IsUsingPsuedonym>

<BirthDate>1985-06-30T00:00:00</BirthDate>

<IsBirthDateAnEstimate>false</IsBirthDateAnEstimate>

<GenderCode>MALE</GenderCode>

<ResidentialAddress>

<Suburb>Kambah</Suburb>

<State>ACT</State>

<Postcode>2905</Postcode>

<AddressLine1>Unit 1</AddressLine1>

<AddressLine2>3 xyz street</AddressLine2>

</ResidentialAddress>

<CountryOfBirthCode>1101</CountryOfBirthCode>

<LanguageSpokenAtHomeCode>1201</LanguageSpokenAtHomeCode>

<AboriginalOrTorresStraitIslanderOriginCode>NO</AboriginalOrTorresStraitIslanderOriginCode>

<HasDisabilities>YES</HasDisabilities>

<HomelessIndicatorCode>true</HomelessIndicatorCode>

<HouseholdCompositionCode>GROUP</HouseholdCompositionCode>

<MainSourceOfIncomeCode>NIL</MainSourceOfIncomeCode>

<IncomeFrequencyCode>FORTNIGHTLY</IncomeFrequencyCode>

<IncomeAmount>1</IncomeAmount>

<FirstArrivalYear>2014</FirstArrivalYear>

<MigrationVisaCategoryCode>SKILLED</MigrationVisaCategoryCode>

<AncestryCode>5208</AncestryCode>

<AccommodationTypeCode>BOARDING</AccommodationTypeCode>

<DVACardStatusCode>GOLD</DVACardStatusCode>

<HasCarer>true</HasCarer>

<EducationLevelCode>PREPRIMARY</EducationLevelCode>

<EmploymentStatusCode>WORKFULLTIME</EmploymentStatusCode>

<IsClientACarer>true</IsClientACarer>

<NDISEligibilityCode>INPROGRESS</NDISEligibilityCode>

<Disabilities>

<DisabilityCode>Physical</DisabilityCode>

</Disabilities>

<CreatedDateTime>2014-09-17T16:05:12.273</CreatedDateTime>

</Client>

<Client>

<ClientId>CL0012</ClientId>

<SLK>LO2OE240220141</SLK>

<ConsentToProvideDetails>true</ConsentToProvideDetails>

<ConsentedForFutureContacts>true</ConsentedForFutureContacts>

<ConsentGivenToDepartment>true</ConsentGivenToDepartment>

<GivenName>Joe</GivenName>

<FamilyName>Blog</FamilyName>

<IsUsingPsuedonym>false</IsUsingPsuedonym>

<BirthDate>2014-02-24T00:00:00</BirthDate>

<IsBirthDateAnEstimate>false</IsBirthDateAnEstimate>

<GenderCode>MALE</GenderCode>

<ResidentialAddress>

<Suburb>Forde</Suburb>

<State>ACT</State>

<Postcode>2914</Postcode>

<AddressLine1/>

<AddressLine2/>

</ResidentialAddress>

<CountryOfBirthCode>1101</CountryOfBirthCode>

<LanguageSpokenAtHomeCode>1201</LanguageSpokenAtHomeCode>

<AboriginalOrTorresStraitIslanderOriginCode>NO</AboriginalOrTorresStraitIslanderOriginCode>

<HasDisabilities>YES</HasDisabilities>

<HomelessIndicatorCode>true</HomelessIndicatorCode>

<HouseholdCompositionCode>GROUP</HouseholdCompositionCode>

<MainSourceOfIncomeCode>NIL</MainSourceOfIncomeCode>

<IncomeFrequencyCode>FORTNIGHTLY</IncomeFrequencyCode>

<IncomeAmount>1</IncomeAmount>

<FirstArrivalYear>2014</FirstArrivalYear>

<MigrationVisaCategoryCode>SKILLED</MigrationVisaCategoryCode>

<AncestryCode>5208</AncestryCode>

<AccommodationTypeCode>BOARDING</AccommodationTypeCode>

<DVACardStatusCode>GOLD</DVACardStatusCode>

<HasCarer>true</HasCarer>

<EducationLevelCode>PREPRIMARY</EducationLevelCode>

<EmploymentStatusCode>WORKFULLTIME</EmploymentStatusCode>

<IsClientACarer>true</IsClientACarer>

<NDISEligibilityCode>INPROGRESS</NDISEligibilityCode>

<Disabilities>

<DisabilityCode>Physical</DisabilityCode>

</Disabilities>

<CreatedDateTime>2014-09-22T16:19:09.317</CreatedDateTime>

</Client>

</Clients>

<TotalCount>2</TotalCount>

</ValidateForDuplicateClientResponse>

### UpdateClient

The **UpdateClient** operation is to be used when an organisation wishes to update cient data already submitted to the Data Exchange in order to update client details. If the validation of the data is unsuccessful, an error will be generated and no record will be updated.

#### UpdateClient request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| ClientId | Mandatory | String | 50 |  | Organisation’s client Id. This is used to identify and update the client record in DEX database.  **ClientId cannot be updated to a new value.** |
| SLK | Conditional Mandatory | String | 14 |  | Must be supplied if ConsentToProvideDetails is false. |
| ConsentToProvideDetails | Mandatory | Boolean |  |  | Client consent for DSS to collect personal information from organisations for storage on the Data Exchange.  If false:  Then GivenName, FamilyName and Address line (if supplied) are saved as blank. |
| ConsentedForFutureContacts | Mandatory | Boolean |  |  | Client consent to future contact for survey / research / evaluation. |
| GivenName | Mandatory | String | 30 |  | Must be supplied if ConsentToProvideDetails is true. |
| FamilyName | Mandatory | String | 30 |  | Must be supplied if ConsentToProvideDetails is true. |
| IsUsingPsuedonym | Mandatory | Boolean |  |  | true = If name specified is a Psuedonym  false = If name specified is not a Psuedonym |
| BirthDate | Mandatory | DateTime |  |  | IsBirthDateAnEstimate is true then it should be in the format of ‘yyyy-01-01’.  Should be the real birth date in the format of ‘yyyy-mm-dd’ if IsBirthDateAnEstimate is false. |
| IsBirthDateAnEstimate | Mandatory | Boolean |  |  | true = the birth date is just an estimate  false = the birth date is the real birth date |
| GenderCode | Mandatory | String |  |  | Must be a valid Gender code retrieved using GetReferenceData operation. |
| ResidentialAddress |  |  |  |  |  |
| Suburb | Mandatory | String | 50 |  | Suburb from residential address of a client. |
| State | Mandatory | String | 3 |  | State from residential address of a client. |
| Postcode | Mandatory | String | 4 |  | Postcode from residential address of a client. |
| AddressLine1 | Optional | String | 180 |  | Street address from residential address of a client. |
| AddressLine2 | Optional | String | 180 |  | Overflow from Address line 1 from residential address of a client. |
| CountryOfBirthCode | Mandatory | String |  |  | Must be a valid Country code retrieved using GetReferenceData operation. |
| LanguageSpokenAtHomeCode | Mandatory | String |  |  | Must be a valid Language code retrieved using GetReferenceData operation. |
| AboriginalOrTorresStraitIslanderOriginCode | Mandatory | String |  |  | Must be a valid AboriginalOrTorresStraitIslanderOrigin code retrieved using GetReferenceData operation. |
| HasDisabilities | Mandatory | Boolean |  |  | If true, then minimum of one Disability code is required.  If false, then no Disability code should be sent. |
| HomelessIndicatorCode | Optional | String |  |  | Must be a valid HomelessIndicatorCode retrieved using GetReferenceData operation. |
| HouseholdCompositionCode | Optional | String |  |  | Must be a valid HousehouldComposition code. Can be retrieved using GetReferenceData operation. |
| MainSourceOfIncomeCode | Optional | String |  |  | Must be a valid MainSourceOfIncomeCode retrieved using GetReferenceData operation. |
| IncomeFrequencyCode | Optional | String |  |  | Must be a valid IncomeFrequency code retrieved using GetReferenceData operation. |
| IncomeAmount | Optional | Int |  |  | Client income. |
| FirstArrivalMonth | Optional | String |  |  | Month Client first arrived in Australia  Full name of the month: January, February, March, April, May, June, July, August, September, October, November, December |
| FirstArrivalYear | Optional | Int |  |  | Year Client first arrived in Australia. |
| MigrationVisaCategoryCode | Optional | String |  |  | Must be a valid MigrationVisaCategoryCode retrieved using GetReferenceData operation. |
| AncestryCode | Optional | String |  |  | Must be a valid AncestryCode retrieved using GetReferenceData operation. |
| AccommodationTypeCode | Optional | String |  |  | Must be a valid AccommodationType retrieved using GetReferenceData operation. |
| DVACardStatusCode | Optional | String |  |  | Must be a valid DVACardStatus retrieved using GetReferenceData operation. |
| HasCarer | Optional | Boolean |  |  | true = If client has a carer  false = If client does not have a carer |
| EducationLevelCode | Optional | String |  |  | Must be a valid EducationLevelCode retrieved using GetReferenceData operation. |
| EmploymentStatusCode | Optional | String |  |  | Must be a valid EmploymentStatusCode retrieved using GetReferenceData operation. |
| IsClientACarer | Optional | Boolean |  |  | true = If client is a carer  false = If client is not a carer |
| NDISEligibilityCode | Optional | String |  |  | Must be a valid NDISEligibilityCode retrieved using GetReferenceData operation. |
| Disabilities | Optional |  |  |  |  |
| DisabilityCode | Mandatory | String |  |  | Must be a valid DisabilityCode value retrieved using GetReferenceData operation. |

#### UpdateClient example request message

<UpdateClientRequest>

<Client>

<!--type: string-->

<ClientId>CL1022</ClientId>

<!--type: string-->

<SLK>LO222211220001</SLK>

<!--type: Boolean-->

<ConsentToProvideDetails>true</ConsentToProvideDetails>

<!--type: Boolean-->

<ConsentedForFutureContacts>true</ConsentedForFutureContacts>

<!--type: string-->

<GivenName>Atiras</GivenName>

<!--type: string-->

<FamilyName>Blog</FamilyName>

<!--type: Boolean-->

<IsUsingPsuedonym>true</IsUsingPsuedonym>

<!--type: dateTime-->

<BirthDate>2003-12-21T00:00:00</BirthDate>

<!--type: Boolean-->

<IsBirthDateAnEstimate>false</IsBirthDateAnEstimate>

<!--type: string-->

<GenderCode>Male</GenderCode>

<ResidentialAddress>

<!--Optional:-->

<!--type: string-->

<Suburb>Plains</Suburb>

<!--type: string-->

<State>NSW</State>

<!--type: string-->

<Postcode>2600</Postcode>

<!--type: string-->

<AddressLine1>12 Abc St</AddressLine1>

<!--Optional:-->

<!--type: string-->

<AddressLine2>Rose wood park</AddressLine2>

</ResidentialAddress>

<!--type: string-->

<CountryOfBirthCode>1101</CountryOfBirthCode>

<!--type: string-->

<LanguageSpokenAtHomeCode>1201</LanguageSpokenAtHomeCode>

<!--type: string-->

<AboriginalOrTorresStraitIslanderOriginCode>NOTSTATED</AboriginalOrTorresStraitIslanderOriginCode>

<!--type: Boolean-->

<HasDisabilities>true</HasDisabilities>

<!--Optional:-->

<!--type: string-->

<HomelessIndicatorCode>ATRISK</HomelessIndicatorCode>

<!--Optional:-->

<!--type: string-->

<HouseholdCompositionCode>SINGLE</HouseholdCompositionCode>

<!--Optional:-->

<!--type: string-->

<MainSourceOfIncomeCode>EMPLOYED</MainSourceOfIncomeCode>

<!--Optional:-->

<!--type: string-->

<IncomeFrequencyCode>WEEKLY</IncomeFrequencyCode>

<!--Optional:-->

<!--type: int-->

<IncomeAmount>300</IncomeAmount>

<!--Optional:-->

<!--type: Month - enumeration: [January,February,March,April,May,June,July,August,September,October,November,December]-->

<FirstArrivalMonth>January</FirstArrivalMonth>

<!--Optional:-->

<!--type: int-->

<FirstArrivalYear>1999</FirstArrivalYear>

<!--Optional:-->

<!--type: string-->

<MigrationVisaCategoryCode>HUMANITARIAN</MigrationVisaCategoryCode>

<!--Optional:-->

<!--type: string-->

<AncestryCode>1103</AncestryCode>

<!--Optional:-->

<!--type: string-->

<AccommodationTypeCode>BOARDING</AccommodationTypeCode>

<!--Optional:-->

<!--type: string-->

<DVACardStatusCode>GOLD</DVACardStatusCode>

<!--Optional:-->

<!--type: Boolean-->

<HasCarer>true</HasCarer>

<!--Optional:-->

<!--type: string-->

<EducationLevelCode>PREPRIMARY</EducationLevelCode>

<!--Optional:-->

<!--type: string-->

<EmploymentStatusCode>WORKFULLTIME</EmploymentStatusCode>

<!--Optional:-->

<!--type: Boolean-->

<IsClientACarer>true</IsClientACarer>

<!--Optional:-->

<!--type: string-->

<NDISEligibilityCode>INPROGRESS</NDISEligibilityCode>

<Disabilities>

<!--Zero or more repetitions:-->

<!--type: string-->

<DisabilityCode>NOTSTATED</DisabilityCode>

</Disabilities>

</Client>

</UpdateClientRequest>

#### UpdateClient response message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |  |

#### UpdateClient response message

<UpdateClientResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</UpdateClientResponse>

### DeleteClient

The **DeleteClient** operation can be used to delete a client registered with the Data Exchange. If the client is attached to any cases then it will not be deleted.

#### DeleteClient request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| ClientId | Mandatory | String |  |  | Organisation’s Client Id. The Client is not deleted if attached to any Cases. |

#### DeleteClient example request message

<DeleteClientRequest>

<!--type: string-->

<ClientId>123456</ClientId>

</DeleteClientRequest>

#### DeleteClient response message definition

If the transaction is successful, the following structure will be returned.

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |  |

#### DeleteClient example response message

<DeleteClientResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</DeleteClientResponse>

## DataCollection.Case

### AddCase

The **AddCase** operation is to be used when an organisation wishes to submit data to the Data Exchange in order to create a case. This operation also allows attaching registered clients to the case being created.

#### AddCase request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| Case |  |  |  |  |  |
| CaseId | Mandatory | String | 30 |  | Organisation’s Case Id.  A CaseId is **not** case sensitive.  A client’s name, part of a client’s name, or other identifiable information should not be used as a CaseId under any circumstances. |
| OutletActivityId | Mandatory | Int |  |  | DEX’s OutletActivityId.  Must be for an active OutletActivity**.** Can be retrieved using GetOutletActivities operation. |
| TotalNumberOfUnidentifiedClients | Conditional  Mandatory | Int |  |  | Number of unidentified clients. Allowed range 0-999. |
| ClientAttendanceProfileCode | Optional | String |  |  | Must be a valid ClientAttendanceProfileCode retrieved using GetReferenceData operation. |
| EndDate | Optional | Date |  |  | End date for specified case. |
| AgBusinessTypeCode | Optional | String |  |  | Must be a valid AgBusinessType code retrieved using GetReferenceData operation.  Optional according to Activity selected in case. |
| ParentingAgreementOutcome | Optional | ParentingAgreementOutcome |  |  |  |
| ParentingAgreementOutcomeCode | Conditional  Mandatory | String |  |  | Must be a valid ParentingAgreement code retrieved using GetReferenceData operation. |
| DateOfParentingAgreement | Conditional  Mandatory | DateTime |  |  | ParentingAgreementOutcome Date. |
| DidLegalPractitionerAssistWithFormalisingAgreement | Conditional  Mandatory | Boolean |  |  | Did a Legal Practitioner assist when Parenting Agreement was formalised (or similar). |
| Section60I | Optional | Section60I |  |  |  |
| Section60ICertificateTypeCode | Conditional  Mandatory | String |  |  | Must be a valid Section60ITypecode retrieved using GetReferenceData operation. |
| DateIssued | Conditional  Mandatory | DateTime |  |  | Date Section60I Certificate is issued. |
| PropertyAgreementOutcome | Optional | PropertyAgreementOutcome |  |  |  |
| PropertyAgreementOutcomeCode | Conditional  Mandatory | String |  |  | Must be a valid PropertyAgreement code retrieved using GetReferenceData operation. |
| DateOfPropertyAgreement | Conditional  Mandatory | DateTime |  |  | PropertyAgreementOutcome Date. |
| DidLegalPractitionerAssistInPropertyMediation | Conditional  Mandatory | Boolean |  |  | Did a legal practitioner assist in mediating property agreement (or similar). |
| Clients | Optional | Array (CaseClient) |  |  | Zero or More organisation client Ids to be associated with the case.  The client should already be registered for the user’s organisation. |
| CaseClient | Conditional  Mandatory |  |  |  |  |
| ClientId | Conditional  Mandatory | String |  |  | Organisation’s client Id.  Mandatory If client is provided.  It has to be unique for all the clients registered with the organisation. |
| ReasonsForAssistance | Optional | Array (ReasonForAssistance) |  |  |  |
| ReasonForAssistance | Optional | ReasonForAssistance |  |  | If one or more reasons are sent, One must indicate one reason as the primary reason for seeking assistance.  The user can send multiple secondary reasons for seeking assistance. |
| AssistanceNeededCode | Conditional  Mandatory | String |  |  | Must be a valid AssistanceNeededCode retrieved using GetReferenceData operation. |
| IsPrimary | Conditional  Mandatory | Boolean |  |  | Specify if this reason is primary or not. |
| ReferralSourceCode | Optional | String |  |  | Must be a valid ReferralSourceCode retrieved using GetReferenceData operation. |
| ExitReasonCode | Optional | String |  |  | Must be a valid ExitReasonCode retrieved using GetReferenceData operation. |

#### AddCase example Rrequest message

<AddCaseRequest>

<Case>

<!--type: string-->

<CaseId>Ca11082014</CaseId>

<!--type: int-->

<OutletActivityId>398</OutletActivityId>

<!--Optional:-->

<!--type: int-->

<TotalNumberOfUnidentifiedClients>3</TotalNumberOfUnidentifiedClients>

<!--Optional:-->

<!--type: string-->

<ClientAttendanceProfileCode>PSGROUP</ClientAttendanceProfileCode>

<!--Optional:-->

<!--type: dateTime-->

<EndDate>2006-08-20T00:00:00</EndDate>

<!--Optional:-->

<!--type: string-->

<AgBusinessTypeCode>0111</AgBusinessTypeCode>

<!--Optional:-->

<ParentingAgreementOutcome>

<!--type: string-->

<ParentingAgreementOutcomeCode>FULL</ParentingAgreementOutcomeCode>

<!--type: dateTime-->

<DateOfParentingAgreement>2003-12-21T00:00:00</DateOfParentingAgreement>

<!--type: Boolean-->

<DidLegalPractitionerAssistWithFormalisingAgreement>true</DidLegalPractitionerAssistWithFormalisingAgreement>

</ParentingAgreementOutcome>

<!--Optional:-->

<Section60I>

<!--type: string-->

<Section60ICertificateTypeCode>GENUINE</Section60ICertificateTypeCode>

<!--type: dateTime-->

<DateIssued>2003-12-21T00:00:00</DateIssued>

</Section60I>

<!--Optional:-->

<PropertyAgreementOutcome>

<!--type: string-->

<PropertyAgreementOutcomeCode>FULL</PropertyAgreementOutcomeCode>

<!--type: dateTime-->

<DateOfPropertyAgreement>2003-12-21T00:00:00</DateOfPropertyAgreement>

<!--type: Boolean-->

<DidLegalPractitionerAssistInPropertyMediation>true</DidLegalPractitionerAssistInPropertyMediation>

</PropertyAgreementOutcome>

</Case>

<Clients>

<!--Zero or more repetitions:-->

<CaseClient>

<!--type: string-->

<ClientId>CL0012</ClientId>

<ReasonsForAssistance>

<!--Zero or more repetitions:-->

<ReasonForAssistance>

<!--type: string-->

<AssistanceNeededCode>PHYSICAL</AssistanceNeededCode>

<!--type: Boolean-->

<IsPrimary>true</IsPrimary>

</ReasonForAssistance>

</ReasonsForAssistance>

<!--type: string-->

<ReferralSourceCode>COMMUNITY</ReferralSourceCode>

<!--Optional:-->

<!--type: string-->

<ExitReasonCode>MOVED</ExitReasonCode>

</CaseClient>

</Clients>

</AddCaseRequest>

#### AddCase response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| TransactionStatus |  |  |  |  |

#### AddCase example response message

<AddCaseResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</AddCaseResponse>

### GetCase

The **GetCase** operation is to be used when an organisation wishes to retrieve case data from the Data Exchange. The operation also retrieves the attached clients and sessions.

#### GetCase request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| CaseId | Mandatory | String |  |  | Organisation’s case Id. |

#### GetCase Example Request Message

<GetCaseRequest>

<!--type:string-->

<CaseId>CA0050</CaseId>

</GetCaseRequest>

#### GetCase response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |
| OutletName | String |  | Organisation’s outlet name. |
| CaseDetail |  |  |  |
| CaseId | String |  | Organisation’s case Id. |
| OutletActivityId | Int |  | DEX’s OutletActivityId. |
| TotalNumberOfUnidentifiedClients | Int |  | Number of unidentified clients. Allowed range 0-999. |
| ClientAttendanceProfileCode | String |  | The ClientAttendanceProfileCode value. |
| EndDate | Date |  | The case end date. |
| AgBusinessTypeCode | String |  | The AgBusinessTypeCode value. |
| ProgramActivityName | String |  | DEX’s activity name. |
| CreatedDateTime | DateTime |  | DateTime case was created in DEX’s database. |
| Clients | Array (CaseClient) |  | List of associated clients. |
| CaseClient | CaseClient |  | Associated client. |
| ClientId | String |  | Organisation’s client Id. |
| ReasonsForAssistance | Array(ReasonForAssistance) |  | List of reasons for assistance for a client. |
| ReasonForAssistance | ReasonForAssistance |  | Reason for assistance for a client. |
| AssistanceNeededCode | String |  | The AssistanceNeededCode value. |
| IsPrimary | Boolean |  | Specify if this reason is primary or not. |
| ReferralSourceCode | String |  | The ReferralSourceCode value. |
| ExitReasonCode | String |  | The ExitReasonCode value. |
| Sessions | Array |  |  |
| SessionId | String |  | List of sessions that are attached to the case. |
| ParentingAgreementOutcome | ParentingAgreementOutcome |  |  |
| ParentingAgreementOutcomeCode | String |  | The ParentingAgreement code value. |
| DateOfParentingAgreement | DateTime |  | ParentingAgreementOutcome Date. |
| DidLegalPractitionerAssistWithFormalising Agreement | Boolean |  | Did a Legal Practitioner assist when parenting agreement was formalised (or similar). |
| Section 60I |  |  |  |
| Section60ICertificateTypeCode | String |  | The Section60ITypecode value. |
| Date Issued | DateTime |  | Date Section60I Certificate is issued. |
| PropertyAgreementOutcome | PropertyAgreementOutcome |  |  |
| PropertyAgreementOutcomeCode | String |  | The PropertyAgreement code value. |
| DateOfPropertyAgreement | DateTime |  | PropertyAgreementOutcome Date. |
| DidLegalPractitionerAssistInPropertyMediation | Boolean |  | Did a legal practitioner assist in mediating property agreement (or similar). |

#### GetCase example response message

<GetCaseResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Case>

<OutletName>AAAAustralianSupportServices</OutletName>

<CaseDetail>

<CaseId>CA0050</CaseId>

<OutletActivityId>946</OutletActivityId>

<TotalNumberOfUnidentifiedClients>28</TotalNumberOfUnidentifiedClients>

<ClientAttendanceProfileCode>PSGROUP</ClientAttendanceProfileCode>

<EndDate>2006-08-20T00:00:00</EndDate>

<AgBusinessTypeCode>0111</AgBusinessTypeCode>

</CaseDetail>

<ProgramActivityName>Children's Contact Services</ProgramActivityName>

<CreatedDateTime>2014-05-14T14:52:05.92</CreatedDateTime>

<Clients>

<CaseClient>

<ClientId>DT7</ClientId>

<ReasonsForAssistance>

<ReasonForAssistance>

<AssistanceNeededCode>MATERIAL</AssistanceNeededCode>

<IsPrimary>true</IsPrimary>

</ReasonForAssistance>

</ReasonsForAssistance>

<ReferralSourceCode>HEALTHAGENCY</ReferralSourceCode

<ExitReasonCode>MOVED</ExitReasonCode>

</CaseClient>

<CaseClient>

<ClientId>079</ClientId>

<ReasonsForAssistance/>

<ReferralSourceCode/>

<ExitReasonCode/>

</CaseClient>

<CaseClient>

<ClientId>078</ClientId>

<ReasonsForAssistance/>

<ReferralSourceCode/>

<ExitReasonCode/>

</CaseClient>

</Clients>

<Sessions>

<SessionId>0001</SessionId>

</Sessions>

<ParentingAgreementOutcome>

<ParentingAgreementOutcomeCode>FULL</ParentingAgreementOutcomeCode>

<DateOfParentingAgreement>2013-071-01T00:00:00</DateOfParentingAgreement>

<DidLegalPractitionerAssistWithFormalisingAgreement>false</DidLegalPractitionerAssistWithFormalisingAgreement>

</ParentingAgreementOutcome>

<Section60I>

<Section60ICertificateTypeCode>GENUINE</Section60ICertificateTypeCode>

<DateIssued>2014-06-01T00:00:00</DateIssued>

</Section60I>

<PropertyAgreementOutcome>

<PropertyAgreementOutcomeCode>FULL</PropertyAgreementOutcomeCode>

<DateOfPropertyAgreement>2013-071-01T00:00:00</DateOfPropertyAgreement>

<DidLegalPractitionerAssistInPropertyMediation>false</DidLegalPractitionerAssistInPropertyMediation>

</PropertyAgreementOutcome>

</Case>

</GetCaseResponse>

### SearchCase

**SearchCase** operation can be used to search for user’s organisation cases based on the specified criteria.

#### SearchCase request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| Search Fields (  Search Request Message Definition) |  |  |  |  |  |
| CaseId | Optional | String | 30 |  | Organisation’s case Id.  Any case matching the CaseId will be retrieved. |
| OutletId | Optional | Int |  |  | DEX’s OutletId. |
| OutletActivityId | Optional | Int |  |  | DEX’s OutletActivityId. |
| CreatedDateFrom | Optional | Date |  |  | If only CreatedDateFrom is specified then CreatedDateTo defaults to Today. |
| CreatedDateTo | Optional | Date |  |  | If only CreatedDateTo is specified then CreatedDateFrom defaults to start of DEX. |
| CreatedByMe | Optional | Boolean |  |  | If **true**, only cases created by current user will be retrieved. |
| ClientExternalId | Optional | String |  |  | Organisation’s client Id.  Any case matching the ClientId will be retrieved. |
| EndDateFrom | Optional | Date |  |  | Look for cases where end date is specified.  Within the date range provided.  If only EndDateFrom is specified then EndDateTo defaults to Today.  Specifying these criteria will exclude all active (without end date) cases. |
| EndDateTo | Optional | Date |  |  | Look for cases where end date is specified.  Within the date range provided.  If only EndDateTo is specified then EndDateFrom defaults to start of DEX.  Specifying these criteria will exclude all active (without end date) cases. |

#### SearchCase example request message

<SearchCaseRequest>

<Criteria>

<!--type:int-->

<PageIndex>1</PageIndex>

<!--type:int-->

<PageSize>3</PageSize>

<!--type:Boolean-->

<IsAscending>false</IsAscending>

<!--type:CaseSearchSortColumns-enumeration:[CreatedDate,CaseId,OutletActivity,Outlet,SessionCount]-->

<SortColumn>Outlet</SortColumn>

<!--Optional:-->

<!--type:string-->

<CaseId>0008</CaseId>

<!--Optional:-->

<!--type:int-->

<OutletId>17</OutletId>

<!--Optional:-->

<!--type:int-->

<OutletActivityId>685</OutletActivityId>

<!--Optional:-->

<!--type:dateTime-->

<CreatedDateFrom>2014-05-01T00:00:00</CreatedDateFrom>

<!--Optional:-->

<!--type:dateTime-->

<CreatedDateTo>2014-05-10T00:00:00</CreatedDateTo>

<!--Optional:-->

<!--type:Boolean-->

<CreatedByMe>true</CreatedByMe>

<!--Optional:-->

<!--type: string-->

<ClientExternalId>clientID123</ClientExternalId>

<!--Optional:-->

<!--type:dateTime-->

<dex:EndDateFrom>2015-09-10T00:00:00</dex:EndDateFrom>

<!--Optional:-->

<!--type:dateTime-->

<dex:EndDateTo>2015-09-15T00:00:00</dex:EndDateTo>

</Criteria>

</SearchCaseRequest>

#### SearchCase response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |
| Cases | Array (CaseSearchRecord) |  |  |
| Case |  |  |  |
| OutletName | String |  | Organisation’s outlet name. |
| CaseDetail |  |  |  |
| CaseId | String |  | Organisation’s case Id. |
| OutletActivityId | Int |  | DEX’s OutletActivityId. |
| TotalNumberOfUnidentifiedClients | Int |  | Number of unidentified clients. Allowed range 0-999. |
| ClientAttendanceProfileCode | String |  |  |
| EndDate | Date |  |  |
| ProgramActivityName | String |  | Activity name. |
| CreatedDateTime | DateTime |  | DateTime Case was created in DEX’s database. |
| SessionCount | Int |  | Number of sessions associated with the case. |

#### SearchCase example response message

<SearchCaseResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Cases>

<Case>

<OutletName>The Roman Catholic Bishop of Geraldton Centacare Family Services</OutletName>

<CaseDetail>

<CaseId>0008</CaseId>

<OutletActivityId>685</OutletActivityId>

<TotalNumberOfUnidentifiedClients>2</TotalNumberOfUnidentifiedClients>

<ClientAttendanceProfileCode>PSGROUP</ClientAttendanceProfileCode>

<EndDate>2018-04-28T00:00:00</EndDate>

</CaseDetail>

<ProgramActivityName>Family Dispute Resolution</ProgramActivityName>

<CreatedDateTime>2014-05-08T10:23:55.3</CreatedDateTime>

<SessionCount>8</SessionCount>

</Case>

</Cases>

<TotalCount>1</TotalCount>

</SearchCaseResponse>

### UpdateCase

The **UpdateCase** operation is to be used when an organisation wishes to updated previously submitted case data to the Data Exchange. This includes adding additional clients, removing clients, etc.

#### UpdateCase example request definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| Case |  |  |  |  |  |
| CaseId | Mandatory | String | 50 |  | Organisation’s case Id. It is used to retrieve and update the Case record in DEX database.  **CaseId cannot be updated to new value.** |
| OutletActivityId | Mandatory | Int |  |  | DEX’s OutletActivityId.  Must be for an active OutletActivity.Can be retrieved using GetOutletActivities operation.  Cannot be updated for cases, which have sessions attached. |
| TotalNumberOfUnidentifiedClients | Mandatory | Int |  |  | Number of unidentified clients. Allowed range 0-999. |
| ClientAttendanceProfileCode | Optional | String |  |  | Must be a valid ClientAttendanceProfileCode retrieved using GetReferenceData operation. |
| EndDate | Optional | Date |  |  | Case end date. |
| AgBusinessTypeCode | Optional | String |  |  | Must be a valid AgBusinessType code retrieved using GetReferenceData operation.  Optional according to activity selected in case. |
| ParentingAgreementOutcome | Optional | ParentingAgreementOutcome |  |  |  |
| ParentingAgreementOutcomeCode | Conditional  Mandatory | String |  |  | Must be a valid ParentingAgreement code retrieved using GetReferenceData operation. |
| DateOfParentingAgreement | Conditional  Mandatory | DateTime |  |  | ParentingAgreementOutcome Date. |
| DidLegalPractitionerAssistWithFormalisingAgreement | Conditional  Mandatory | Boolean |  |  | Did a legal practitioner assist when parenting agreement was formalised (or similar). |
| Section60I | Optional | Section60I |  |  |  |
| Section60ICertificateTypeCode | Conditional  Mandatory | String |  |  | Must be a valid Section60ICertificateTypecode retrieved using GetReferenceData operation. |
| DateIssued | Conditional  Mandatory | DateTime |  |  | Date Section60I Certificate is issued. |
| PropertyAgreementOutcome | Optional | PropertyAgreementOutcome |  |  |  |
| PropertyAgreementOutcomeCode | Conditional  Mandatory | String |  |  | Must be a valid PropertyAgreement code retrieved using GetReferenceData operation. |
| DateOfPropertyAgreement | Conditional  Mandatory | DateTime |  |  | PropertyAgreementOutcome date. |
| DidLegalPractitionerAssistInPropertyMediation | Conditional  Mandatory | Boolean |  |  | Did a Legal Practitioner assist in mediating property agreement (or similar). |
| Clients | Optional | Array (String) |  |  | Zero or More organisation CaseClient to be associated with the Case.  The client should already be registered for the user’s organisation.  The **existing** attached CaseClients, which are not passed in the request, will be **removed**. |
| CaseClient | Conditional  Mandatory | String |  |  |  |
| ClientId | Conditional  Mandatory | String |  |  | Organisation’s client Id.  Mandatory if client is provided  It has to be unique for all the clients registered with the organisation. |
| ReasonsForAssistance | Optional | Array(ReasonForAssistance) |  |  |  |
| ReasonForAssistance | Optional | ReasonForAssistance |  |  | If one or more reasons are sent, One must indicate one reason as the primary reason for seeking assistance.  The user can send multiple secondary reasons for seeking assistance. |
| AssistanceNeededCode | Conditional  Mandatory | String |  |  | Must be a valid AssistanceNeededCode retrieved using GetReferenceData operation. |
| IsPrimary | Conditional  Mandatory | Boolean |  |  | Specify if this reason is primary or not. |
| ReferralSourceCode | Optional | String |  |  | Must be a valid ReferralSourceCode retrieved using GetReferenceData operation. |
| ExitReasonCode | Optional | String |  |  | Must be a valid ExitReaspmCode retrieved using GetReferenceData operation. |

#### UpdateCase example request message

<UpdateCaseRequest>

<Case>

<!--type:string-->

<CaseId>UTCa11082014</CaseId>

<!--type:int-->

<OutletActivityId>634</OutletActivityId>

<!--type:int-->

<TotalNumberOfUnidentifiedClients>3</TotalNumberOfUnidentifiedClients>

<!--type:string-->

<ClientAttendanceProfileCode>PSGROUP</ClientAttendanceProfileCode>

<!--type: dateTime-->

<EndDate>2018-04-28T00:00:00</EndDate>

<!--type:string-->

<AgBusinessTypeCode>0111</AgBusinessTypeCode>

<!--Optional:-->

<ParentingAgreementOutcome>

<!--type: string-->

<ParentingAgreementOutcomeCode>FULL</ParentingAgreementOutcomeCode>

<!--type: dateTime-->

<DateOfParentingAgreement>2003-12-21T00:00:00</DateOfParentingAgreement>

<!--type: Boolean-->

<DidLegalPractitionerAssistWithFormalisingAgreement>true</DidLegalPractitionerAssistWithFormalisingAgreement>

</ParentingAgreementOutcome>

<!--Optional:-->

<Section60I>

<!--type: string-->

<Section60ICertificateTypeCode>GENUINE</Section60ICertificateTypeCode>

<!--type: dateTime-->

<DateIssued>2003-12-21T00:00:00</DateIssued>

</Section60I>

<!--Optional:-->

<PropertyAgreementOutcome>

<!--type: string-->

<PropertyAgreementOutcomeCode>FULL</PropertyAgreementOutcomeCode>

<!--type: dateTime-->

<DateOfPropertyAgreement>2003-12-21T00:00:00</DateOfPropertyAgreement>

<!--type: Boolean-->

<DidLegalPractitionerAssistInPropertyMediation>true</ DidLegalPractitionerAssistInPropertyMediation>

</PropertyAgreementOutcome>

<Clients>

<!--Zeroormorerepetitions:-->

<CaseClient>

<!--type:string-->

<ClientId>132</ClientId>

<ReasonsForAssistance>

<!--Zeroormorerepetitions:-->

<ReasonForAssistance>

<!--type:string-->

<AssistanceNeededCode>MENTAL</AssistanceNeededCode>

<!--type:Boolean-->

<IsPrimary>true</IsPrimary>

</ReasonForAssistance>

</ReasonsForAssistance>

<!--type:string-->

<ReferralSourceCode>Legal</ReferralSourceCode>

<!--Optional:-->

<!--type: string-->

<ExitReasonCode>MOVED</ExitReasonCode>

</CaseClient>

</Clients>

</UpdateCaseRequest>

#### UpdateCase response message definition

If the transaction is successful, the following structure will be returned

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### UpdateCase example response message

<UpdateCaseResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</UpdateCaseResponse>

### DeleteCase

The **DeleteCase** operation can be used to delete a case created with the Data Exchange. If the case has any sessions attached, then it will **not** be deleted.

#### DeleteCase request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| CaseId | Mandatory | String |  |  | Organisation’s Case Id. |

#### DeleteCase example request message

<DeleteCaseRequest>

<!--type:string-->

<CaseId>A0050</CaseId>

</DeleteCaseRequest>

#### DeleteCase response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### DeleteCase example response message

<DeleteCaseResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</DeleteCaseResponse>

## DataCollection.Session

### AddSession

**AddSession** operation can be used by the organisation to add new sessions in a case.

#### AddSession request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| CaseId | Mandatory |  |  |  | Organisation’s case Id.  Must be an existing Case that is not locked. |
| Session | Mandatory |  |  |  |  |
| SessionId | Mandatory | String |  |  | Organisation’s session Id.  Must be unique for all the Sessions in a Case. It is **not** case sensitive.  A client’s name, part of a client’s name, or other identifiable information should not be used as a SessionId under any circumstances. |
| SessionDate | Mandatory | DateTime |  |  | Date of the session.  Must be today or in past and within OutletActivity date range. |
| ServiceTypeId | Mandatory | Int |  |  | DEX’s ServiceTypeId.  Can be retrieved using GetOrganisationActivities. |
| TotalNumberOfUnidentifiedClients | Conditional Mandatory | Int |  |  | Cannot exceed Case. TotalNumberOfUnidentifiedClients. |
| FeesCharged | Optional | Decimal |  |  | Fees Charged to Client for selective activities. |
| MoneyBusinessCommunityEducationWorkshopCode | Optional | String |  |  | Must be a valid. MoneyBusinessCommunityEducationWorkshopCode retrieved using GetReferenceData operation.  Optional according to Activity and Service type. |
| InterpreterPresent | Optional | Boolean |  |  | Specifying where interpreter was present for session. |
| ExtraItems | Optional | Array (ExtraItem) |  |  | Optional according to activity and service type. |
| ExtraItemCode | Optional | String |  |  | Must be a valid ExtraItemCode retrieved using GetReferenceData operation. |
| Quantity | Optional | Int |  |  | Optional according to Activity and Service type. |
| Time | Optional | Int |  |  | Optional according to Activity and Service type. |
| TotalCost | Optional | Int |  |  | Optional according to Activity and Service type. |
| TopicCode | Optional | String |  |  | Must be a valid TopicCode retrieved using GetReferenceData operation.  Optional according to activity and service type. |
| ServiceSettingCode | Optional | String |  |  | Must be a valid ServiceSettingCode retrieved using GetReferenceData operation. |
| HardshipTypeCode | Optional | String |  |  | Must be a valid HardshipTypeCode retrieved using GetReferenceData operation.  Optional according to activity and service type. |
| ExternalReferralDestinationCode | Optional | String |  |  | Must be a valid ExternalReferralDestinationCode retrieved using GetReferenceData operation.  Optional according to activity and service type. |
| Clients | Optional | Array (SessionClient) |  |  | Zero or more clients which are associated with the Case. |
| SessionClient | Optional |  |  |  |  |
| ClientId | Conditional Mandatory | String |  |  | Organisation’s client Id.  Mandatory If client is provided.  It has to be unique for all the clients registered with the organisation. |
| ParticipationCode | Conditional Mandatory | String |  |  | Mandatory If Client is provided.  Must be a valid ParticipationCode retrieved using GetReferenceData operation. |
| ClientReferralOutWithPurpose | Optional | Array (Referral) |  |  | List of client referrals. |
| Referral | Optional |  |  |  | Client referral at the session. |
| TypeCode | Conditional Mandatory | String |  |  | Mandatory If referral is provided.  Must be a valid TypeCode retrieved using GetReferenceData operation. |
| PurposeCodes | Conditional Mandatory | Array (string) |  |  | List Of DEX’s referral purpose codes.  Mandatory If referral is provided. |
| PurposeCode | Conditional Mandatory | String |  |  | Mandatory If referral is provided.  Must be a valid PurposeCode retrieved using GetReferenceData operation. |

#### AddSession example request message

<AddSessionRequest>

<!--typestring-->

<CaseId>UTCa11082014</CaseId>

<Session>

<!--typestring-->

<SessionId>UTSe11082014</SessionId>

<!--typedateTime-->

<SessionDate>2014-05-10T190103+1100</SessionDate>

<!--typeint-->

<ServiceTypeId>22</ServiceTypeId>

<!--Optional-->

<!--typeint-->

<TotalNumberOfUnidentifiedClients>3</TotalNumberOfUnidentifiedClients>

<!--Optional-->

<!--typedecimal-->

<FeesCharged>1000.00</FeesCharged>

<!--Optional-->

<!--typestring-->

<MoneyBusinessCommunityEducationWorkshopCode>WRK01</MoneyBusinessCommunityEducationWorkshopCode>

<!--Optional-->

<!--typeboolean-->

<InterpreterPresent>true</InterpreterPresent>

<!--Optional-->

<ExtraItems>

<!--Zeroormorerepetitions-->

<!--typestring-->

<ExtraItemCode>KITCHEN</ExtraItemCode>

</ExtraItems>

<!--Optional-->

<!--typeint-->

<Quantity>1</Quantity>

<!--Optional-->

<!--typeint-->

<Time>30</Time>

<!--Optional-->

<!--typeint-->

<TotalCost>100</TotalCost>

<!--Optional-->

<!--typestring-->

<TopicCode>ABUSENEGLECT</TopicCode>

<!--Optional-->

<!--typestring-->

<ServiceSettingCode>COMMVENUE</ServiceSettingCode>

<!--Optional-->

<!--typestring-->

<HardshipTypeCode>DROUGHT</HardshipTypeCode>

<!--Optional-->

<!--typestring-->

<ExternalReferralDestinationCode>LEGAL</ExternalReferralDestinationCode>

<Clients>

<!--Zeroormorerepetitions-->

<SessionClient>

<!--typestring-->

<ClientId>CL0050</ClientId>

<!--typestring-->

<ParticipationCode>Client</ParticipationCode>

<!--Optional-->

<ClientReferralOutWithPurpose>

<!—Zero or more repetitions-->

<Referral>

<!--typestring-->

<TypeCode>Internal</TypeCode>

<PurposeCodes>

<!--Zeroormorerepetitions-->

<!--typestring-->

<PurposeCode>PERSONAL</PurposeCode>

</PurposeCodes>

</Referral>

</ClientReferralOutWithPurpose>

</SessionClient>

</Clients>

</Session>

</AddSessionRequest>

#### AddSession response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### AddSession example response message

<AddSessionResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</AddSessionResponse>

### GetSession

**GetSession** operation can be used to retrieve details of a session within a case. The operation also returns the clients attached to the session.

#### GetSession request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| CaseId | Mandatory | String | 30 |  | Organisation’s case Id. |
| SessionId | Mandatory | String | 30 |  | Organisation’s session Id. |

#### GetSession example request message

<GetSessionRequest>

<!--type:string-->

<CaseId>CA00010</CaseId>

<!--type:string-->

<SessionId>SS00010</SessionId>

</GetSessionRequest>

#### GetSession response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |
| CaseId |  |  | Organisation’s case Id. |
| SessionDetails |  |  |  |
| SessionId | String |  | Organisation’s session Id |
| SessionDate | DateTime |  | Date of the session. |
| ServiceTypeId | Int |  | DEX’s ServiceTypeId. |
| TotalNumberOfUnidentifiedClients | Int |  | Cannot exceed Case TotalNumberOfUnidentifiedClients. |
| FeesCharged | Decimal |  | Fees Charged to Client for selective activities. |
| MoneyBusinessCommunityEducationWorkshopCode | String |  | The MoneyBusinessCommunityEducationWorkshopCode value. |
| InterpreterPresent | Boolean |  | Specifying where interpreter was present for a session. |
| ExtraItems | Array (ExtraItem) |  | Optional according to activity and service type. |
| ExtraItemCode | String |  | The ExtraItemCode value. |
| Quantity | Int |  | Optional according to activity and service type. |
| Time | Int |  | Optional according to activity and service type. |
| TotalCost | Int |  | Optional according to activity and service type. |
| TopicCode | String |  | Optional according to activity and service type. |
| ServiceSettingCode | String |  | The ServiceSettingCode value. |
| HardshipTypeCode | String |  | Optional according to activity and service type. |
| ExternalReferralDestinationCode | String |  | Optional according to activity and service type. |
| CreatedDateTime | DateTime |  | Session created on. |
| Clients | Array (SessionClient) |  | Zero or more clients which are associated with the Session. |
| SessionClient | SessionClient |  |  |
| ClientId | String |  | Organisation’s client Id. |
| ParticipationCode | String |  | The ParticipationCode value. |
| ClientReferralOutWithPurpose | Array (Referral) |  | List of client referrals. |
| Referral | Referral |  | Client referral at the session. |
| TypeCode | String |  | The TypeCode value. |
| PurposeCodes | String |  | List Of DEX’s referral purpose codes. |
| PurposeCode | String |  | The PurposeCode value. |
| ClientAssessments | Array (Assessment) |  | List of client assessments. |
| Assessment | Assessment |  | Client assessment. |
| AssessmentPhaseCode | String |  | The AssessmentPhaseCode value. |
| ScoreTypeCode | String |  | The ScoreTypeCode value. |
| AssessedByCode | String |  | The AssessedByCode value. |
| Scores | Array (string) |  | List of scores. |
| ScoreCode | String |  | The ScoreCode value. |
| SessionAssessments | Array (Assessment) |  | List of session sssessments. |
| Assessment | Assessment |  | Session assessments. |
| AssessmentPhaseCode | String |  | The AssessmentPhaseCode value. |
| ScoreTypeCode | String |  | The ScoreTypeCode value. |
| AssessedByCode | String |  | The AssessedByCode value. |
| Scores | Array (string) |  | List of scores. |
| ScoreCode | String |  | The ScoreCode value. |

#### GetSession example response message

<GetSessionResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<Session>

<CaseId>CA00010</CaseId>

<SessionDetail>

<SessionId>SS00010</SessionId>

<SessionDate>2014-05-30T00:00:00</SessionDate>

<ServiceTypeId>12</ServiceTypeId>

<TotalNumberOfUnidentifiedClients>2</TotalNumberOfUnidentifiedClients>

<FeesCharged>5.00</FeesCharged>

<MoneyBusinessCommunityEducationWorkshopCode>WRK01</MoneyBusinessCommunityEducationWorkshopCode>

<InterpreterPresent>true</InterpreterPresent>

<ExtraItems>

<ExtraItemCode>KITCHEN</ExtraItemCode>

</ExtraItems>

<Quantity>1</Quantity>

<Time>30</Time>

<TotalCost>100</TotalCost>

<TopicCode>ABUSENEGLECT</TopicCode>

<ServiceSettingCode>COMMVENUE</ServiceSettingCode>

<HardshipTypeCode>DROUGHT</HardshipTypeCode>

<ExternalReferralDestinationCode>LEGAL</ExternalReferralDestinationCode>

</SessionDetails>

<CreatedDateTime>2014-09-17T16:19:14.32</CreatedDateTime>

<Clients>

<SessionClient>

<ClientId>CLB00015</ClientId>

<ParticipationCode>CLIENT</ParticipationCode>

<ClientReferralOutWithPurpose>

<Referral>

<TypeCode>EXTERNAL</TypeCode>

<PurposeCodes>

<PurposeCode>HOUSING</PurposeCode>

<PurposeCode>PHYSICAL</PurposeCode>

</PurposeCodes>

</Referral>

<Referral>

<TypeCode>INTERNAL</TypeCode>

<PurposeCodes>

<PurposeCode>HOUSING</PurposeCode>

</PurposeCodes>

</Referral>

</ClientReferralOutWithPurpose>

<ClientAssessments>

<Assessment>

<AssessmentPhaseCode>Pre</AssessmentPhaseCode>

<ScoreTypeCode>Circumstances</ScoreTypeCode>

<Scores>

<ScoreCode>Age3</ScoreCode>

</Scores>

</Assessment>

</ClientAssessments>

</SessionClient>

<SessionClient>

<ClientId>CLB00016</ClientId>

<ParticipationCode>CLIENT</ParticipationCode>

<ClientReferralOutWithPurpose>

<Referral>

<TypeCode>INTERNAL</TypeCode>

<PurposeCodes>

<PurposeCode>HOUSING</PurposeCode>

<PurposeCode>PHYSICAL</PurposeCode>

</PurposeCodes>

</Referral>

</ClientReferralOutWithPurpose>

<ClientAssessments>

<Assessment>

<AssessmentPhaseCode>PRE</AssessmentPhaseCode>

<ScoreTypeCode>CIRCUMSTANCES</ScoreTypeCode>

<AssessedByCode>SDJOINT</AssessedByCode>

<Scores>

<ScoreCode>AGE1</ScoreCode>

</Scores>

</Assessment>

</ClientAssessments>

</SessionClient>

</Clients>

<SessionAssessments>

<Assessment>

<AssessmentPhaseCode>PRE</AssessmentPhaseCode>

<ScoreTypeCode>GROUP</ScoreTypeCode>

<AssessedByCode>SDJOINT</AssessedByCode>

<Scores>

<ScoreCode>ORGSKILLS1</ScoreCode>

</Scores>

</Assessment>

</SessionAssessments>

</Session>

</GetSessionResponse>

### UpdateSession

The **UpdateSession** operation is to be used when an organisation wishes to updated previously submitted Session data to the Data Exchange. This includes adding additional clients, removing clients, etc.

#### UpdateSession request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| CaseId | Mandatory | String |  |  | Organisation’s case Id.  Must be an existing Case that is not locked. |
| Session |  |  |  |  |  |
| SessionId | Mandatory | String | 30 |  | Organisation’s session Id.  Session Id and Case Id are used to identify a unique Session record, which is updated.  **SessionId cannot be updated to new value.** |
| SessionDate | Mandatory | DateTime |  |  | Date of the session.  Must be today or in past and within OutletActivity date range. |
| ServiceTypeId | Mandatory | Int |  |  | DEX’s serviceTypeId.  Can be retrieved using GetOrganisationActivities. |
| TotalNumberOfUnidentifiedClients | Mandatory | Int |  |  | Cannot exceed TotalNumberOfUnidentifiedClients in a Case. |
| FeesCharged | Optional | Decimal |  |  | Fees charged to client for selective activities. |
| MoneyBusinessCommunityEducationWorkshopCode | Optional | String |  |  | Must be a valid. MoneyBusinessCommunityEducationWorkshopCode retrieved using GetReferenceData operation. |
| InterpreterPresent | Optional | Boolean |  |  | Specifying where interpreter was present for a session. |
| ExtraItems | Optional | Array (ExtraItem) |  |  | Optional according to activity and service type. |
| ExtraItemCode | Optional | String |  |  | Must be a valid ExtraItemCode retrieved using GetReferenceData operation. |
| Quantity | Optional | Int |  |  | Optional according to activity and service type. |
| Time | Optional | Int |  |  | Optional according to activity and service type. |
| TotalCost | Optional | Int |  |  | Optional according to activity and service type. |
| TopicCode | Optional | String |  |  | Must be a valid TopicCode retrieved using GetReferenceData operation.  Optional according to activity and service type. |
| ServiceSettingCode | Optional | String |  |  | Must be a valid ServiceSettingCode retrieved using GetReferenceData operation. |
| HardshipTypeCode | Optional | String |  |  | Must be a valid HardshipTypeCode retrieved using GetReferenceData operation.  Optional according to activity and service type. |
| ExternalReferralDestinationCode | Optional | String |  |  | Must be a valid ExternalReferralDestinationCode retrieved using GetReferenceData operation.  Optional according to activity and service type. |
| Clients | Optional | Array (SessionClient) |  |  | Zero or more clients which are associated with the Case. |
| SessionClient | Optional |  |  |  |  |
| ClientId | Conditional Mandatory | String |  |  | Organisation’s client Id.  Mandatory If client is provided.  It has to be unique for all the clients registered with the organisation. |
| ParticipationCode | Conditional Mandatory | String |  |  | Mandatory If client is provided.  Must be a valid ParticipationCode retrieved using GetReferenceData operation. |
| ClientReferralOutWithPurpose | Optional | Array (Referral) |  |  | List of client Referrals. |
| Referral | Optional |  |  |  | Client referral at the session. |
| TypeCode | Conditional Mandatory | String |  |  | Mandatory If referral is provided.  Must be a valid TypeCode retrieved using GetReferenceData operation. |
| PurposeCodes | Conditional Mandatory | Array (string) |  |  | List Of DEX’s referral purpose codes.  Mandatory If Referral is provided. |
| PurposeCode | Conditional Mandatory | String |  |  | Mandatory If referral is provided.  Must be a valid PurposeCode retrieved using GetReferenceData operation. |

#### UpdateSession example request message

<UpdateSessionRequest>

<!--typestring-->

<CaseId>0050</CaseId>

<Session>

<!--typestring-->

<SessionId>0001</SessionId>

<!--typedateTime-->

<SessionDate>2014-03-12T000000</SessionDate>

<!--typeint-->

<ServiceTypeId>5</ServiceTypeId>

<!--typeint-->

<TotalNumberOfUnidentifiedClients>3</TotalNumberOfUnidentifiedClients>

<!--Optional-->

<!--typedecimal-->

<FeesCharged>1000.00</FeesCharged>

<!--Optional-->

<!--typestring-->

<MoneyBusinessCommunityEducationWorkshopCode>WRK01</MoneyBusinessCommunityEducationWorkshopCode>

<!--typeboolean-->

<InterpreterPresent>true</InterpreterPresent>

<ExtraItems>

<!--Zeroormorerepetitions-->

<!--typestring-->

<ExtraItemCode>KITCHEN</ExtraItemCode>

</ExtraItems>

<!--Optional-->

<!--typeint-->

<Quantity>1</Quantity>

<!--Optional-->

<!--typeint-->

<Time>30</Time>

<!--Optional-->

<!--typeint-->

<TotalCost>100</TotalCost>

<!--Optional-->

<!--typestring-->

<TopicCode>ABUSENEGLECT</TopicCode>

<!--Optional-->

<!--typestring-->

<ServiceSettingCode>COMMVENUE</ServiceSettingCode>

<!--Optional-->

<!--typestring-->

<HardshipTypeCode>DROUGHT</HardshipTypeCode>

<!--Optional-->

<!--typestring-->

<ExternalReferralDestinationCode>LEGAL</ExternalReferralDestinationCode>

<Clients>

<!--Zeroormorerepetitions-->

<SessionClient>

<!--typestring-->

<ClientId>CL0050</ClientId>

<!--typestring-->

<ParticipationCode>SUPPORT</ParticipationCode>

<!--Optional-->

<ClientReferralOutWithPurpose>

<!--Zeroormorerepetitions-->

<Referral>

<!--typestring-->

<TypeCode>INTERNAL</TypeCode>

<PurposeCodes>

<!--Zeroormorerepetitions-->

<!--typestring-->

<Purpose>AGE</Purpose>

</PurposeCodes>

</Referral>

</ClientReferralOutWithPurpose>

</SessionClient>

</Clients>

</Session>

</UpdateSessionRequest>

#### UpdateSession response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### UpdateSession example response message

<UpdateSessionResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</UpdateSessionResponse>

### DeleteSession

**DeleteSession** operation can be used to delete a Session attached to a Case.

#### DeleteSession request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| CaseId | Mandatory | String |  |  | Organisation’s case Id. |
| SessionId | Mandatory | String |  |  | Organisation’s session Id for the session to be deleted. |

#### DeleteSession example request message

<DeleteSessionRequest>

<!--typestring-->

<CaseId>0050</CaseId>

<!--typestring-->

<SessionId>0001</SessionId>

</DeleteSessionRequest>

#### DeleteSession response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### DeleteSession example response message

<DeleteSessionResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</DeleteSessionResponse>

## DataCollection.Assessments

### UpdateClientAssessments

The **UpdateClientAssessments** operation is to be used when an organisation wishes to add new client assessment data, or update cient assessment data already submitted to the Data Exchange. If the validation of the data is unsuccessful, an error will be generated and no record will be updated.

#### UpdateClientAssessments request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| ClientAssessment |  |  |  |  |  |
| CaseId | Mandatory | String |  |  | Organisation’s case Id.  Must be an existing Case that is not locked. |
| SessionId | Mandatory | String |  |  | Organisation’s session Id for the Session to be updated. |
| ClientId | Mandatory | String |  |  | Client Id who attended the session to be updated. |
| Assessments | Optional |  |  |  | List of new or/and updated assessments for a client at a session. |
| Assessment | Optional |  |  |  | New or/and updated assessments for a client at a session. |
| AssessmentPhaseCode | Mandatory | String |  |  | Must be a valid AssessmentPhaseCode retrieved using GetReferenceData operation. |
| ScoreTypeCode | Mandatory | String |  |  | Must be a valid ScoreTypeCode retrieved using GetReferenceData operation. |
| AssessedByCode | Optional | String |  |  | Must be a valid AssessedByCode retrieved using GetReferenceData operation |
| Scores | Mandatory | Array(string) |  |  | List of score codes for assessment. |
| ScoreCode | Mandatory | String |  |  | Must be a valid ScoreCode retrieved using GetAssessmentReferenceDetails operation. |
| RemoveAll | Mandatory | Boolean |  |  | Confirmation flag to indicate to remove all the assessment records associated with the client assessment. |

#### UpdateClientAssessments example request message

<UpdateClientAssessmentsRequest>

<ClientAssessment>

<!--type:string-->

<CaseId>CA0005A</CaseId>

<!--type:string-->

<SessionId>SS0123</SessionId>

<!--type:string-->

<ClientId>SA0002</ClientId>

<Assessments>

<!--Zeroormorerepetitions:-->

<Assessment>

<!--type:string-->

<AssessmentPhaseCode>PRE</AssessmentPhaseCode>

<!--type:string-->

<ScoreTypeCode>GOALS</ScoreTypeCode>

<!--type:string-->

<AssessedByCode>SDJOINT</AssessedByCode>

<Scores>

<!--Zeroormorerepetitions:-->

<!--type:string-->

<ScoreCode>BEHAVIOURS3</ScoreCode>

</Scores>

</Assessment>

</Assessments>

<!--type:Boolean-->

<RemoveAll>false</RemoveAll>

</ClientAssessment>

</UpdateClientAssessmentsRequest>

#### UpdateClientAssessments response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### UpdateClientAssessments example response message

<UpdateClientAssessmentsResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</UpdateClientAssessmentsResponse>

### UpdateSessionAssessments

The **UpdateSessionAssessments** operation is to be used when an organisation wishes to add new session assessment data associated with the session, or update session assessment data associated with the session that is already submitted to the Data Exchange. If the validation of the data is unsuccessful, an error will be generated and no record will be updated.

#### UpdateSessionAssessments request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| SessionAssessment |  |  |  |  |  |
| CaseId | Mandatory | String |  |  | Organisation’s case Id.  Must be an existing Case that is not locked. |
| SessionId | Mandatory | String |  |  | Organisation’s Session Id for the session to be updated. |
| Assessments | Optional |  |  |  | List of new or/and updated assessments for a session applicable to all clients in a session. |
| Assessment | Optional |  |  |  | New or/and updated assessments for a session applicable to all Clients in a Session. |
| AssessmentPhaseCode | Mandatory | String |  |  | Must be a valid AssessmentPhaseCode retrieved using GetReferenceData operation.  Maximum 1 pre and 1 post assessment per session. |
| ScoreTypeCode | Mandatory | String |  |  | Must be a valid ScoreTypeCode retrieved using GetReferenceData operation. |
| AssessedByCode | Optional | String |  |  | Must be a valid AssessedByCode retrieved using GetReferenceData operation. |
| Scores | Optional |  |  |  | List Of score codes for assessment. |
| ScoreCode | Mandatory | String |  |  | Must be a valid ScoreCode retrieved using GetAssessmentReferenceDetails operation. |
| RemoveAll | Mandatory | Boolean |  |  | Confirmation flag to indicate to remove all the assessment records associated with the session assessment for the case. |

#### UpdateSessionAssessments example request message

<UpdateSessionAssessmentsRequest>

<SessionAssessment>

<!--type:string-->

<CaseId>CA0005A</CaseId>

<!--type:string-->

<SessionId>SS0123</SessionId>

<Assessments>

<!--Zeroormorerepetitions:-->

<Assessment>

<!--type:string-->

<AssessmentPhaseCode>Pre</AssessmentPhaseCode>

<!--type:string-->

<ScoreTypeCode>Group</ScoreTypeCode>

<!--type:string-->

<AssessedByCode>SDJOINT</AssessedByCode>

<Scores>

<!--Zeroormorerepetitions:-->

<!--type:string-->

<ScoreCode>ORGSKILLS1</ScoreCode>

</Scores>

</Assessment>

</Assessments>

<!--type:Boolean-->

<RemoveAll>false</RemoveAll>

</SessionAssessment>

</UpdateSessionAssessmentsRequest>

#### UpdateSessionAssessments response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### UpdateSessionAssessments example response message

<UpdateSessionAssessmentsResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</UpdateSessionAssessmentsResponse>

## DataCollection.Outlets

### AddOutlet

The **AddOutlet** operation is to be used when an organisation wishes add a new outlet to the Data Exchange system. Outlet name and address are mandatory. Appropriate **OutletActivities** can be included at the same time the outlet is created. While adding a new outlet, **OutletActivities** are optional.

However, while performing an **UpdateOutlet** operation, new **OutletActivities** can be added or existing ones updated, but any **OutletActivities** that are omitted from the update will indicate that they are to be removed from the **Outlet** within the Data Exchange.

If the validation of the data is unsuccessful, an error will be generated and no record will be updated.

#### AddOutlet Request Message Definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| OutletDetails |  |  |  |  |  |
| OutletName | Mandatory | String |  |  |  |
| Address | Mandatory |  |  |  |  |
| Suburb | Mandatory | String | 50 |  | Suburb from address of an outlet. |
| State | Mandatory | String | 3 |  | State from address of an outlet – sourced from GetReferenceData – State. |
| Postcode | Mandatory | String | 4 |  | Postcode from address of an outlet. |
| AddressLine1 | Mandatory | String | 180 |  | Street address from address of an outlet. |
| AddressLine2 | Optional | String | 180 |  | Overflow from address line 1 from address of an outlet. |
| OutletActivities | Optional |  |  |  |  |
| OutletActivity |  |  |  |  |  |
| OrganisationActivityId | Mandatory | Int |  |  | Sourced from GetOrganisationActivities. |
| StartDate | Mandatory | Date |  |  | Date range must be equal to or within the OrganisationActivity start and end dates. |
| EndDate | Mandatory | Date |  |  | Date range must be equal to or within the OrganisationActivity start and end dates. |

#### AddOutlet example request message

<AddOutletRequest>

<OutletDetails>

<OutletName>Bob’s Family Services</OutletName>

<Address>

<Suburb>Springfield</Suburb>

<State>ACT</State>

<Postcode>2600</Postcode>

<AddressLine1>1 Main Street</AddressLine1>

<AddressLine2/>

</Address>

<OutletActivities>

<OutletActivity>

<OrganisationActivityId>324</OrganisationActivityId>

<StartDate>2015-07-01T00:00:00+10:00</StartDate>

<EndDate>2015-12-31T00:00:00+10:00</EndDate>

</OutletActivity>

</OutletActivities>

</OutletDetails>

</AddOutletRequest>

#### AddOutlet response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |
| OutletId | Int |  | The OutletId of the successfully added outlet. |

#### AddOutlet Example response message

<AddOutletResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

<OutletId>999</OutletId>

</AddOutletResponse>

### UpdateOutlet

The **UpdateOutlet** operation is to be used when an organisation wishes update an Outlet that exists in the Data Exchange system. OutletId, Outlet name and address are mandatory.

While performing an UpdateOutlet operation, new **OutletActivities** can be added or existing ones updated, but any **OutletActivities** that are omitted from the update will indicate that they are to be removed from the Outlet within the Data Exchange. An **OutletActivity** will only be deleted if no Case data is associated with it, otherwise it will be end dated to the current date.

If the validation of the data is unsuccessful, an error will be generated and no record will be updated.

#### UpdateOutlet request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| UpdateOutletRequest |  |  |  |  |  |
| OutletId | Mandatory | Int |  |  | Id of an Outlet added to DEX via AddOutlet. Can be sourced from GetOrganisation details. |
| OutletName | Mandatory | String |  |  |  |
| Address | Mandatory |  |  |  |  |
| Suburb | Mandatory | String | 50 |  | Suburb from address of an outlet. |
| State | Mandatory | String | 3 |  | State from address of an outlet – sourced from GetReferenceData – State. |
| Postcode | Mandatory | String | 4 |  | Postcode from address of an outlet. |
| AddressLine1 | Mandatory | String | 180 |  | Street address from address of an outlet. |
| AddressLine2 | Optional | String | 180 |  | Overflow from address line 1 from address of an outlet. |
| OutletActivities | Optional |  |  |  |  |
| OutletActivity |  |  |  |  | NOTE: Omitting an OutletActivity that has previously been added to the Outlet will indicate that the OutletActivity should be removed/deleted. |
| OrganisationActivityId | Mandatory | Int |  |  | Sourced from GetOrganisationActivities. |
| StartDate | Mandatory | Date |  |  | Date range must be equal to or within the OrganisationActivity start and end dates. |
| EndDate | Mandatory | Date |  |  | Date range must be equal to or within the OrganisationActivity start and end dates. |

#### UpdateOutlet example request message

<UpdateOutletRequest>

<OutletId>435</OutletId>

<OutletDetails>

<OutletName>Bob’s Family Services</OutletName>

<Address>

<Suburb>Springfield</Suburb>

<State>ACT</State>

<Postcode>2600</Postcode>

<AddressLine1>1 Main Street</AddressLine1>

<AddressLine2/>

</Address>

<OutletActivities>

<OutletActivity>

<OrganisationActivityId>324</OrganisationActivityId>

<StartDate>2015-07-01T00:00:00+10:00</StartDate>

<EndDate>2015-12-31T00:00:00+10:00</EndDate>

</OutletActivity>

</OutletActivities>

</OutletDetails>

</UpdateOutletRequest>

#### UpdateOutlet response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### UpdateOutlet example response message

<UpdateOutletResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</UpdateOutletResponse>

### DeleteOutlet

The **DeleteOutlet** operation is to be used when an organisation wishes delete an outlet that exists in the Data Exchange system.

The outlet will be deleted only when there is no case data associated with it. **OutletActivities** with no case data will be deleted, otherwise **OutletActivities** with case data will be end dated to the current date and the parent Outlet will not be deleted.

Warning messages will be returned indicating if the outlet cannot be deleted due to related case data.

#### DeleteOutlet Request message definition

| Field | Mandatory Conditional Optional | Data Type | Length | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- | --- | --- |
| OutletId | Mandatory | Int |  |  | Id of an Outlet added to DEX via AddOutlet. Can be sourced from GetOrganisation details. |

#### DeleteOutlet example request message

<DeleteOutletRequest>

<OutletId>435</OutletId>

</DeleteOutletRequest>

#### DeleteOutlet response message definition

If the transaction is successful, the following structure will be returned.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| TransactionStatus |  |  |  |

#### DeleteOutlet example response message

<DeleteOutletResponse>

<TransactionStatus>

<TransactionStatusCode>Success</TransactionStatusCode>

<Messages/>

</TransactionStatus>

</DeleteOutletResponse>

## Common message definitions

### Request message definitions

#### Search request message definition

The following fields are common to **SearchClient**, **SearchCase** and **ValidateForDuplicateClient** operations. The fields are mandatory and must be passed in the requests along with the other operation specific fields described in the relevant sections below.

| Field | Mandatory Conditional Optional | Data Type | Length | Comments Validation Rules |
| --- | --- | --- | --- | --- |
| PageIndex | Mandatory | Int |  | The page number. |
| PageSize | Mandatory | Int |  | Number of records returned per page.  The max value allowed is **100**. If value greater than 100 is passed, then the result set is truncated to 100 records and a warning message is returned. |
| IsAscending | Mandatory | Boolean |  | Acceptable values: true, false. |
| SortColumn | Mandatory | Enumeration |  | The following are valid **SortColumn** values for the **SearchClient** and **ValidateForDuplicateClient**:   * CreatedDate * ClientId * Name * BirthDate   The following are valid **SortColumn** values for the **SearchCase**:   * CreatedDate * CaseId * OutletActivity * Outlet * SessionCount |

### Response message definitions

#### Client response message definition

The following client response message definition fields are common to **GetClient**, **SearchClient** and some of the other operations.

| Field | Data Type | Reference Date | Comments Validation Rules |
| --- | --- | --- | --- |
| Client |  |  |  |
| ClientDetail |  |  |  |
| ClientId | String |  | Organisation’s client Id. |
| SLK | String |  | Client SLK generated or supplied . |
| ConsentToProvideDetails | Boolean |  | Personal details consent. |
| ConsentedForFutureContacts | Boolean |  | Future contact consent. |
| GivenName | String |  | Clients given name. |
| FamilyName | String |  | Clients family name. |
| IsUsingPsuedonym | Boolean |  | Specifying if name is a psuedonym or not. |
| BirthDate | DateTime |  | Date Of Birth. |
| IsBirthDateAnEstimate | Boolean |  | Specifying if BirthDate is an estimate or not. |
| GenderCode | String |  | GenderCode value. |
| ResidentialAddress |  |  | Client address details. |
| Suburb | String |  | Suburb from residential address of a client. |
| State | String |  | State from residential address of a client. |
| Postcode | String |  | Postcode from residential address of a client. |
| AddressLine1 | String |  | Street address from residential address of a client. |
| AddressLine2 | String |  | Overflow from address line 1 from residential address of a client. |
| CountryOfBirthCode | String |  | CountryOfBirthCode value. |
| LanguageSpokenAtHomeCode | String |  | LanguageSpokenAtHomeCode value. |
| AboriginalOrTorresStraitIslanderOriginCode | String |  | AboriginalOrTorresStraitIslanderOriginCode value. |
| HasDisabilities | Boolean |  | Specifying if Client has disabilities. |
| HomelessIndicatorCode | String |  | HomelessIndicatorCode value. |
| HouseholdCompositionCode | String |  | HouseholdCompositionCode value. |
| MainSourceOfIncomeCode | String |  | MainSourceOfIncomeCode value. |
| IncomeFrequencyCode | String |  | IncomeFrequencyCode value. |
| IncomeAmount | Int |  | Client IncomeAmount . |
| FirstArrivalMonth | String |  | Month Client first arrived in Australia  Full name of the month:  January, February, March, April, May, June, July, August, September, October, November, December |
| FirstArrivalYear | Int |  | Year client first arrived in Australia. |
| MigrationVisaCategoryCode | String |  | MigrationVisaCategoryCode value. |
| AncestryCode | String |  | AncestryCode value. |
| HasCarer | Boolean |  | true = If client has a carer  false = If client does not have a carer |
| EducationLevelCode | String |  | EducationLevelCode value. |
| EmploymentStatusCode | String |  | EmploymentStatusCode value. |
| IsClientACarer | Boolean |  | true = If client is a carer  false = If client is not a carer |
| NDISEligibilityCode | String |  | NDISEligibilityCode value. |
| Disabilities | Array |  |  |
| DisabilityCode | String |  | DisabilityCode value. |
| CreatedDateTime | DateTime |  | DateTime Client was created in DEX’s database. |

# Appendix A. Activity specific requirements

The following lists the activity specific requirements fields.

| Code | Related to entity and property | Description |
| --- | --- | --- |
| ACCOMMODATIONTYPE | Client: AccommodationTypeCode | If a client is associated to a case where its activity specifies this code, then ‘AccommodationTypeCode’ property on the client is required and must be supplied with a valid value. This also implies that clients will not be accepted on a session within that case. |
| DVACARDSTATUS | Client: DVACardStatusCode | If a Client is associated to a case where its activity specifies this code, then ‘DVACardStatusCode’ property on the Client is required and must be supplied with a valid value. This also implies that Clients will not be accepted on a session within that case. |
| HOUSEHOLDMANDATORY | Client: HouseholdCompositionCode | If a client is associated to a case where its activity specifies this code, then ‘HouseholdCompositionCode’ property on the client is required and must be supplied with a valid value. This also implies that clients will not be accepted on a session within that case. |
| HASCARER | Client: HasCarer | If a client is associated to a case where its activity specifies this code, then ‘HasCarer’ property on the client is required and must be supplied with a valid value. This also implies that clients will not be accepted on a session within that case. |
| PARENTINGAGREEMENTOUTCOME | Case:  ParentingAgreementOutcome: ParentingAgreementOutcomeCode, DateOfParentingAgreement, DidLegalPractitionerAssistWithFormalisingAgreement | When applied to an activity, ‘ParentingAgreementOutcomeCode’, ‘DateOfParentingAgreement’ and ‘DidLegalPractitionerAssistWithFormalisingAgreement’ properties on case can be supplied. |
| CERTIFICATETYPE | Case:  Section60I: Section60ICertificateTypeCode, DateIssued | When applied to an activity, ‘Section60ICertificateTypeCode’ and ‘DateIssued’ properties on case can be supplied. |
| PROPERTYAGREEMENTOUTCOME | Case:  PropertyAgreementOutcome: PropertyAgreementOutcomeCode, DateOfPropertyAgreement, DidLegalPractitionerAssistInPropertyMediation | When applied to an activity, ‘PropertyAgreementOutcomeCode’, ‘DateOfPropertyAgreement’ and ‘DidLegalPractitionerAssistInPropertyMediation’ properties on Case can be supplied. |
| AGBUSINESSTYPE | Case: AgBusinessTypeCode | When applied to an activity, ‘AgBusinessTypeCode’ property for Case can be supplied. |
| EXITCASE | Case Client: ExitReasonCode | When applied to an activity, ‘ExitReasonCode’ property for the Case Client can be supplied. |
| FEESCHARGEDSESSIONMANDT | Session: FeesCharged | When applied to an activity, ‘FeesCharged’ property on Session must be supplied and with a valid value. |
| FEESCHARGEDSESSIONOPTNL | Session: FeesCharged | When applied to an activity, ‘FeesCharged’ property on Session can be supplied. If supplied, must have a valid value. |
| TOPIC | Session: TopicCode | When applied to an activity, ‘TopicCode’ property on Session must be supplied and with a valid value. |
| UNIDENTIFIEDCLIENTS | Case and Session | When applied to an Activity, ‘TotalNumberOfUnidentifiedClients’ property on case and session can be supplied and is required. When supplied, must have a valid value. |

# Appendix B. Service type specific requirements

List of applicable codes for a service type is available from **GetOrganisationActivities** operation.

| **Code** | **Related entity and property** | **Description** |
| --- | --- | --- |
| ASSISTANCECOSTAMOUNT | Session: TotalCost | When applied to a service type ‘TotalCost’ property on session is required and must be supplied with valid values to record a new session or update an existing session. |
| ASSISTANCEITEM | Session:  ExtraItems: ExtraItemCode | When applied to a service type ‘ExtraItems’ property on session is required and must be supplied with a valid value to record a new session or update an existing session. |
| ASSISTANCEMINUTES | Session: Time | When applied to a service type ‘Time’ property on session is required and must be supplied with a valid value to record a new session or update an existing session. |
| ASSISTANCEQUANTITY | Session: Quantity | When applied to a service type ‘Quantity’ property on session is required and must be supplied with a valid value to record a new session or update an existing session. |
| WORKSHOP | Session: MoneyBusinessCommunityEducationWorkshopCode | When applied to a service type ‘MoneyBusinessCommunityEducationWorkshopCode’ property on session is required and must be supplied with a valid value to record a new session or update an existing session. |
| HARDSHIPTYPE | Session: HardshipTypeCode | When applied to a service type ‘HardshipTypeCode’ property on session is required and must be supplied with a valid value to record a new session or update an existing session. |
| EXTERNALREFERRALDESTINATION | Session: ExternalReferralDestinationCode | When applied to a service type ‘ExternalReferralDestinationCode’ property on session can be supplied with a valid value to record a new session or update an existing session. |

1. Generally, available means that DSS will endeavour to have the interface available, outages, maintenance periods and factors outside of DSS control may prevent this. [↑](#footnote-ref-1)