Iridium Satellite LLC (“Iridium”)

SOAP Developer Guide

BETA Environment

Last Modified: May 29, 2019

V19.1.6.0

NOTICE

Iridium Proprietary and Confidential

This Developer Guide (the “Guide”) is in “BETA” form. Iridium expressly disclaims all warranties of any kind, whether express or implied, including, but not limited to any warranties of merchantability, fitness for a particular use or purpose, non-infringement, title, accuracy of data and system integration. The Guide may not be accurate and may be substantially modified or withdrawn. The entire risk arising out of the use of the Guide remains with you, the Iridium partner. In no event shall Iridium be liable for any damage whatsoever arising out of the use of or inability to use the Guide, even if Iridium has been advised of the possibility of such damages. Iridium is not obligated to provide updates, maintenance or technical support to the Iridium partner. In no event shall Iridium be obligated to provide to partner a copy of the final version of the Guide nor is Iridium obligated to make the Guide commercially available. In no event shall Iridium be liable for any indirect, incidental, special, consequential or exemplary damages, arising out of or related to your use of the Guide, even if Iridium has been advised of the possibility of such damages. The aggregate liability of Iridium to you for all claims arising from or related to the Guide is limited to amounts paid by you to Iridium in connection therewith. If you are dissatisfied with any portion of the Guide, your sole and exclusive remedy is to discontinue use of the Guide.

BETA

SOAP Developer Guide

1 Table of Contents

BETA

BETA

BETA

BETA

BETA

BETA

BETA

BETA

BETA

BETA

BETA

BETA

Revision History

Date	Version	**Description**	Author
5/29/2019	19.1.6.0	• Update request values for bulkAddPrepaid and bulkRechargePrepaidAccount	IWS Support
5/28/2019	19.1.5.0	• Update vesselLink object with additional information used on output of Vessel • Added planId field to bulkRechargePrepaidAccount • Added status field to CertusScratchcardOrder • Added scratchcardOrderStatusEnum	IWS Support
5/13/2019	19.1.0.0	• New DIDWW calls addDIDWWNumberInventory, getDIDWWNumber, and removeDIDWWNumberInventory added • New bulk prepaid calls bulkAddPrepaid and bulkRechargePrepaidAccount added • New calls for Certus scratchcards getScratchcardDeals, getScratchcardOrders, and orderScratchcards added • Modified activateScratchCards to support Certus functionality • New calls for Vessel support createVessel, getVessels, and updateVessel added • New call getProjectVersion added • Update simplifiedModel and model with new fields for updated Certus provisioning • Added deviceStatusTypeEnum to deviceNumber • Modified deviceApn object to include additional validation fields. • Replaced maritimeSafetyInfo object fields with Vessel object • Update Telephony rudicsGroup field from single field to array of RudicsGroup objects • Update certusImei object with additional model information • Change Certus Streaming structure field from enum to int • Added sdfRatingId to secondaryDataService object • Added deviceApnSearchParrameters to getDevices call • Added imeiSimLockPriority and prepaidDataEnabled fields to spServiceConfigurationInfo object • Added SFX_STANDARD to modelServiceCategoryEnum • Added GMDSS to submarketTypeEnum • Removed following calls:	IWS Support

BETA

BETA

Date	Version	**Description**	Author
		o sendMessageToDevice o getAccountTransitionReport	
9/04/2018	2.7.5	• Added deviceUsageTypeEnum for SP profile MSISDN prioritization • Added iwsLogTypeEnum for iwsRequestLog calls • Updated associateSimToImeiDevice for bulk request options • Updated associateSimToImeiDevice response to return bulk ID for bulk request • Added deviceUsageType as a field to ServiceProviderProfile in findServiceProviderProfile • New calls getIwsRequests, getIwsRequest, unquarantineDevice added • Added CertusImeiImpl to CertusSubscriberAccount object • Compression fields added to primaryDataService object • Removed the following calls: o getAccountTransitionReport o getPrepaidCallHistory o getRealTimeCallDataReport o getResourceInventoryReport o getSubscriberBaseReport o getTransactionHistoryReport o sbdImeiSearch	IWS Support
5/23/2018	2.7.4	• Added ENUM value to sbdDestSearchAndReplaceResultTypeEnum for EMAIL destination swaps on RUSSIA contracts	IWS Support
3/28/2018	2.7.3	• Add Certus Secondary Data Service support o Operation createSecondaryDataService added o Operation updateSecondaryDataService added o Operation getSecondaryDataServices added o Operation getSecondaryDataService added o Operation deleteSecondaryDataService added • validateDeviceString object imeiProfileImpl added serviceCategory object to response. • Added primaryDataService object to certusSubscriberAccount • Added optionalSecondaryDataService object to certusSubscriberAccount • Added bundleOption array to absMemberService object for standalone bundle support on Certus	IWS Support

Date	Version	**Description**	Author
		• Added requireTimeExtension flag to voucherDenominationImpl • Added submarketId field to telephonyBulkPlanUpdate and pttDeviceBulkPlanUpdate • Added optionBundleGroups and bundleOptions objects to getServicePlan response object • Changed permittedServices in getServicePlan response from purchaseConditionEnum to permittedServiceImpl. o Added object permittedServiceImpl • Added new filter servicePlanType for getServicePlans • ENUMs Added o modelServiceCategoryEnum, vesselTypeEnum, primaryDataServiceTypeEnum, secondaryDataServiceOverrideEnum, secondaryDataServiceBillingIdEnum, certusSecondaryDataServicePriorityEnum, servicePlanTypeEnum • ENUMs Updated o queeuEntryStatusEnum, prepaidAccountTypeEnum, subscirberAccountSearchTypeEnum • ENUMs Removed o subscriberBaseRptTypeEnum, resourceInventoryTypeEnum • Deprecated Calls o getRealTimeCallDataReport o m2mImeiSearch o sbdImeiSearch	
12/13/2017	2.7.2	• Added homeGateway to validateDeviceString response object.	Product Management
10/29/2017	2.7.1	• Added NOT_APPLICABLE to businessTypeEnum • Added IHANDSET, ISBD, ILBT, IWIFI to modelServiceClassEnum • Added OTHER to modelServiceCategoryEnum • Added SLBT4, SLBT5, BBT3, BBT4 to dataClassmarkEnum	IWS Support
09/27/2017	2.7.0	• Added support for multiple SBD gateways o New accountUpdateHomeGateway call o Addition of imeiSimBasedFlag to objects: simplifiedModelImpl, imeiProfileImpl, modelImpl	IWS Support

BETA

Date	Version	**Description**	Author
		o Addition of homeGateway to objects: sbdSubscriberAccountImpl, m2MSubscriberAccountImpl, certusSubscriberAccountImpl, accountUpdateDevicesRequestImpl o Additions to spServiceConfigurationInfoImpl for defaultHomeGateway and gatewayOverrideFlag o New ENUM homeGatewayEnum o Add network field to sbdMOQueuePurge and sbdMOQueueReport to identify which gateway queue should be displayed • Removed BASE_64 as an option for sbdSendMessage (messageContentTypeEnum). Send message is intended for testing devices; encrypted messages are no longer a necessary feature. • Added STREAMING to callServiceCodeEnum for Certus usage reporting.	
05/01/2017	2.6.0	• Removed deprecated calls and related objects: o Operation getRoles removed. o Operation revokeRole removed. o Operation getProvisioningFailureReport removed. o Operation grantRole removed. o roleTypeEnum removed o contactTypeEnum removed o serviceOrderStatusEnum removed. • New methods created: o Operation resetIWSUserPassword added. o Operation getSPNetGroups added. o Operation getQueueEntries added. o Operation getQueueEntry added. o Operation getSPNetPages added. o Operation updateQueueEntry added. o Operation disAssociateSimFromImeiDevice added • ENUMs updated: o Added: spNetGroupStatusEnum, operationStatusEnum, endpointEnum, queueEntryStatusEnum, userStatusEnum, outageStatusEnum • validateDeviceString added reason to response for invalid results, update potential result types.	IWS Support

BETA

Date	Version	**Description**	Author
		• Remove deprecated report calls no longer supported: getCallSummaryReport, getResourceInventoryReport, getSubscriberBaseReport, getTransactionHistoryReport, getPrepaidCallHistory • Updated UserImpl object for new user model • Added new attributes to accountSearch Response: plan name, ICCID, IMEI, and spReference values • Added missing documentation for deletePrepaidAccount, getAccountPoolingGroup method. • Updated response object of getSystemStatus to detail new attributes regarding specific components currently disabled, as well as future outages and their affected components. • Added requestType to bulkActionSearch	
01/10/2017	2.5.1	• TelephonyPrePaidPlanImpl renamed to PrePaidPlanImpl. • telephonyCrewPlanImpl renamed to CrewPlanImpl. • Add LOCALNUMBER to deviceNumberTypeEnum • Add airlineGroupId to getAirlinesResponseImpl • Add additional model attributes to getModelResponseImpl • getDeviceUsageReportResponseImpl: o Add spAccountNumber and spName o Update type of initialBalance and currentBalance to int o Change deactivationDate to voucherStatus o Add MSISDN to voucherActivityImpl o Rename duration to callDuration o Rename unitsConsumed to ratedUnits o Add countryName, lac, latMin, latPol, longMin, longPol, tZIndicator, tzOffset, setAltitude, callingHC, callingVC, release Cause o Removed locationImpl • Add enableLocalNumber and localCountry to telephonySubscriberAccountBaseImpl and telephonyBulkBaseImpl **for future use** with expanded local number capabilities • Add localNumberModel to deviceNumberImpl • Add numberModel to localNumImpl to help distinguish different local number regions.	IWS Support
09/01/2016	2.5.0	• Add support for Certus services	IWS Support

BETA

Date	Version	**Description**	Author
		• Refactor GDB calls and rename to Burst o subscribeGDBService → subscribeBurstService o getGDBDeviceBundles → getBurstDeviceBundles o getGDBSubscriptionNames → getBurstSubscriptionNames o getGDBBroadcastCoverageAreas → getBurstBroadcastCoverageAreas o unsubscribeGDBService → unsubscribeBurstService o Rename GLOBAL_DATA_BROADCAST → BURST o Remove GLOBAL_DATA_BROADCAST_SUB_ACCOUNTS, device subscriptions now part of device account structure. • New Calls added o getModels o getModel o getManufacturers o loadIMEI o getVerticals o getLanguageChoices o getLocalNumberRegions o restrictedSimSwap • Added support for MSISDN-B device types primarily for Russian based telephony contracts. • Added account status INCOMPLETE for contracts where primary service is active, but secondary services are in error. • deliveryMethodTypeEnum – Removed SBD_DEVICE and M2M_DEVICE, replaced by IRIDIUM_DEVICE • serviceOrderStatusEnum: INCOMPLETE added. • deviceTypeEnum: LOCAL_NUMBER added. • subscriberAccountSearchTypeEnum: o Removed: SBD_IMEI, M2M_IMEI, GDB_IMEI, PTT_IMEI o Added: MSISDNB, MSISDNC, LOCAL, MMSI, BURST_SERVICE, DELIVERY_DEST, MT_FILTER, MODEL, and VERTICAL. • submarketTypeEnum – Added SAFETY type. • telephonyLanguageTypeEnum renamed to languageTypeEnum. • New Enum types added:	

BETA

Date	Version	**Description**	Author
		o objectStatusEnum, businessTypeEnum, , certusLineTypeEnum, lineQualityEnum, streamingSpeedEnum, networkEnum, secondarySvcTypeEnum, secondarySvcStatusEnum, dateRangeTypeEnum, searchCondEnum, modelServiceCategoryEnum, serviceProviderProfileConfigTypeEnum, modelServiceClassEnum, dataClassmarkEnum, purchaseConditionEnum, serviceLanguageTypeEnum • Add toLocalCountry for local number swap actions in deviceSwapImpl • deviceSimImpl – added submarketType • deviceNumberImpl – added submarketType • imeiProfileImpl – added • accountSearch refactored: o new request object: accountSearchRequestImpl o new response object accountSearchResponseImpl • ResourceInventoryReport – removed model, network. Added serviceType, resType, vertical, manufacturer, modelNum, modelName. • ServicePLanImpl: added serviceCategories, permittedServices • BurstDeviceAccountImpl – added array of subscriptions • BurstServiceAccountImpl – added array of devices • getAccountPoolingGroups – added sp acct number, sp name, service type to response • getServicePlans – added model to request to enable filter result by plans available for specific model. • getaccountTransitionReport –refactored request and response.	
05/19/2016	2.4.2	• Add msisdnb field to result set for telephonySubscriberAccountBaseImpl – currently only used for Russia Gateway support. • Change GLOBAL_DATA_BROADCAST_SUB_ACCOUNTS to GLOBAL_DATA_BROADCAST_SUBSCRIPTION in serviceTypeEnum • Add MSISDNB to deviceNumberTypeEnum	Product Management

BETA

Date	Version	**Description**	Author
04/11/2016	2.4.1	• Added specifics on connection pool limits.	Product Management
02/12/2016	2.4.0	• Upgrade code base to Java 8. • Add requestType as optional parameter for getBulkActionReportRequestImpl • Change stats enum for findUserProfilesRequestImpl and findUserProfilesResponseImpl to userStatusEnum, added LOCKED and EXPIRED status values • Added passwordExpiry date to userImpl to indicate when a user password will expire.	IWSSupport
12/29/2015	2.3.2	• Add more details values to BulkActionEntryImpl. • Add requestType to bulkActionSearchRequestImpl • Add subscriberOrganization to subscriberAccountEntryImpl results. • Add validIpDeviceData, validIcaoData, and validTailNumData optional types to validateDeviceStringResponseImpl • Update messageid type to long in sbdDeleteDownloadResultEntryImpl • Update messageId type to string in sbdDeletePendingDownloadsRequestImpl • Add ATS_ICAO and ATS_TAIL_NUMBER to deviceStringTypeEnum	IWSSupport
11/24/2015	2.3.1	• Updated searchDeviceUsage, ID type long to accommodate larger values, added destination to sbdmoActivityImpl.	IWSSupport
11/06/2015	2.3.0	• 2015 Release II modifications o Add userUnlock call. o Add toDeviceVpn parameter to deviceSwapImpl object o Add requestType to subscriberAccountEntryImpl to enhance search options for bulk action report searches. o Add GDB Service Account details, and GDBService Plan details	IWSSupport
10/04/2015	2.3.0 Draft	• 2015 Release II modifications o Add resetUserPassword call. o accountStatusEnum new value PREPAID_ERROR to indicate the account is valid in billing, but the prepaid side is in error.	IWSSupport
06/02/2015	2.2.0	• 2015 Release 1 modifications	IWSSupport

Date	Version	**Description**	Author
		o Add model to resourceInventoryReportImpl. o Add new fields (trackingEnabled, reportingEnabled), to serviceProviderImpl for tracking and reporting features. o Add imsi and msisdn fields to callSummaryReportImpl, total amount changed to string type. o Add new method resetTelephonyVMPin o Add enum choices for resourceInventoryTypeEnum: PLUS1, PLUS7, NUMBERS, MSISDNC o Add service type as part of criteria for subscriber account search. Add generic subscriberAccountSearchTypeEnum IMEI to be used with new service_type option. Service specific values SBD_IMEI, GDB_IMEI, M2M_IMEI, PTT_IMEI will be removed next release. o Add deactivationDate to the response in subscriberBasereportImpl o Document updatePTTSubOrgStatus method	
12/11/2014	2.1.9	• Add deviceTransferOwner function o Add isDeviceTransfer Boolean to bulk action search request • Enhanced telephony plan functionality o Add servicePlanType to servicePlanImpl o Remove AccountPoolingGroup from telephony base plan o Add submarket as an optional search parameter for to getAccountPoolingGroups request • Add submarket as an optional search parameter for to resourceInventory request • Add IPADDR as new type to deviceStringTypeEnum • Add GLOBAL_DATA_BROADCAST_SUB_ACCOUNTS to serviceTypeEnum	Product Management
11/07/2014	2.1.8	• Add PTT TalkgroupID to pttTalkGroupSubscriberAccountImpl • Modified pttDeviceDetailImpl and pttTalkgroupDetailImpl status fields to be of type accountStatusEnum • Added output fields to subscriberBaseReportImpl: talkGrpId, subscrOrg, twostageVoice,	Product Management

BETA

Date	Version	**Description**	Author
		twostageVoicePin, twostageData, twostageDataPin, atsAirline, atsReg, atsIcao • Add planName output fields to callSummaryReportImpl • Add talkGrpId and subscrOrg output fields to transactionHistoryReportImpl	
10/01/2014	2.1.7	• Release 2 2014 changes applied • Update rechargeHistory response values to include old and new balance as well as old and new expiration dates. • Add createSubscriberOrg • Add getSubscriberOrgs • Add getsubscriberOrgDetail • Add updateSubscriberOrg • Add pttTalkgroup and pttDevice account and plan objects • Update imeiProfileImpl to remove ID field. • Add pttTypeEnum, update serviceTypeEnum and subscriberAccountSearchTypeEnum values for new PTT types. • Add PTT bulk device add and update plan.	Product Management
06/08/2014	2.1.6	• Change contactInfo in userImpl to be required. • Change elements of contactInfo to be required. • Add entry points for getProvisioningFailureReport • Add isPrepaidActive to getServicePlansRequestImpl • Add DemoTrial indicator to servicePlan response. • Add serviceType to bulkActionSearchRequest • Add serviceType to searchAccountRequest • Add new serviceOrderStatusEnum	Product Management
04/08/2014	2.1.5	• Added dependentDeal to service plan options object. • Added fromPlanId to getServicePlans request. • Changed MSISDN field for getPrepaidAccountDetail to required. • Added atsInfoEntry to subscriberAccountEntryImpl for account searches. • Add rudicsGroup, twoStageVoice and twoStageData for telephonyBulkBaseImpl • Changed ATS Primary and Secondary accounts from telephonySubscriberAccountBaseImpl to telephonySubscriberAccountImpl • Changed SIM from deviceSimImpl to string for telephonyBulkDetails object. • Add unitExpiration and unitExpirationDate fields to getPrepaidAccountDetailImpl	Product Management

BETA

Date	Version	**Description**	Author
		• Changed subscriberItem field for getprepaidAccountSummary to required.	
2/24/14	2.1.4	• Added getPrepaidPlans • Added soapAction=OperationName to all methods in the WSDL. • Removed iccid, servicePlanId from rechargePrepaidAccount call • Add forActivate for getServicePlans • Relabel restrict to restrictScratchCards in prepaidAccountDetail • Add prepaidAccountDetail to telephony Crew Plan, remove language and restrictScratchCards (as they are part of prepaidAccountDetail) • Change servicePlanId to servicePlan (type:string) in getPrepaidAccountsummary • Added poolGroup to subscriberBasereport results • Remove prepaidPlan from servicePlanImpl • Added ISDNA to resourceInventoryTypeEnum • Added message to IWSException and authorizationException responses • Changed enabled type from Boolean to string in actionNotificationImpl • Changed size from string to long in sbdActivityImpl • Update telephony bulk parameters.	Product Management
1/8/2013	2.1.3	• Modified return values included in voucherDenominationImpl • Added bulkActionSearch method to retrieve a list of bulk actions. • Added isDemoTrial parameter to getServicePlansRequest • Added getDemoAndTrialCounts method to retrieve specific demo and trial available counts • datTypeEnum value type NA changed to NONE • Added accountUpdateDevices method for device swaps. Swaps on telephony or paging accounts will no longer be supported through accountUpdate action. • Removed ICCID from getPrepaidAccountDetail request. • Subscriber Base Report: • Renamed singlestage fields to plusNumberSubscriber Base Report: • Added legacyContract identifier to assist with migration mappings (will be provided for telephony, paging and SBD accounts)	Product Management

BETA

Date	Version	**Description**	Author
		• Added twostageblock to servicePlan to indicate if twostage options are processed for accounts on the plan. • Added back searchVoucherOrder, voucherOrderSearchTypeEnum, voucherInventoryTypeEnum which were removed in error. • Add prepaidPlan identifier to getServicePlans and getservicePlan responses • Add action Type to getEmailNotificationresponse, and updateEmailNotification request. Add emailNotificationTypeEnum. •	
12/10/2013	2.1.2	• Added displayColumn, displayOrder and requires fields to servicePlanOptions • Added m2mIMEISearch • Added sbdMOQueuePurge • Added sbdMOQueueReport • sbdImeiTypeEnum changed to ImeiTypeEnum and is valid for SBD and M2M • Added new search types for legacy services • getServicePlans modified to return results within submarket array • submarketImpl added id, absServicePlanImpl array • servicePlan added optionalGroups removed paymentModel • submarketCode changed to submarketId • submarketTypeEnum values adjusted • Added plusNumber to TelephonySubscriberAccount, removed singlestageNumber from Telephony Plan • Removed Promo, Demo and Commitment bundles from telephony and Paging plan, promo and commitment will be handled by service option deals. • Removed Telephony and Paging plan objects from bundle requests. • Changed datTypeEnum Demo type NA to NONE.	Product Management
11/05/2013	2.1.1	• Added telephonyLanguageTypeEnum • Added prepaidLanguageTypeEnum • Added searchOperatorEnum • Added groups to servicePlan and serviceOptions • Add more details to accountPoolingGroupTelephony and Paging language changed to string instead of bundle.	Product Management

BETA

Date	Version	**Description**	Author
		• Updated required elements/fields for reusable objects including Account, Plan, Bulk Details, Bulk Plan Update, Device/Number, Prepaid Account Detail and Prepaid Voucher. • Removed Voucher Order Line Item section. • Added “Prepaid” and “ATS” to serviceTypeEnum • Added submarketTypeEnumUpdated activateTelephonyScratchcards response with additional fields. • Deleted service type from deacticateTelephonyScratchcard request and updated the response with additional fields. • Updated deletePrepaidAccount responseUpdated required fields for getAccountTransitionReport, getCallSummaryReport, getRealTimeCallDataReport, getResourceInventoryReport, getSubscriberBaseReport and getTransactionHistoryReport to match the WSDL. • Added getPrepaidAccountDetail callUpdated getPrepaidAccountSummary, getPrepaidCallHistory, getRechargeHistory, getScratchCardBatches, getScratchCardsUpdated getRudicsGroups with serviceType. • Updated getSubmarkets response with paymentModel and submarketType. • Updated getVoucherDenominations request with additional “forActivate” field. • Updated required fields for sbdDeletePendingDownloads, sbdDestinationSearchAndReplace, sbdImeiSearch, searchDeviceUsage, searchSimLock and subscribeGDBServiceUpdated updateEmailNotification with action field in the requestUpdated response fields for getPrepaidCallHistory to reflect required fields. • Updated getRechargeHistory to reflect MSISDN and SIM as required fields. • Updated prepaidAccountDetail with additional fields; modified request to reflect required fields. • Updated getScratchcards with additional fields; modified required fields. • Updates applied to getScratchcards including addition of count of records to response. Deleted from deactivateTelephonyScratchcard, “service types”	

Date	Version	**Description**	Author
		• Updated hyperlinks with correct references	
09/30/2013	2.1.0 Early Adopter	• Updates applied for Telephony/ Paging/Prepaid integration. • Added following sections: • How to Activate a telephony device • How to Activate a paging device • deviceImeiTypeEnum • prepaidAccountTypeEnum • activateTelephonyScratchCards • deactivateTelephonyScratchCard • deletePrepaidAccount • getEmailNotification • getEvoucherInventory • getPrepaidAccountSummary • getPrepaidCallHistory • getRechargeHistory • getServicePlan • getServicePlans • getScratchcardBatches • getScratchcards • getSubmarkets • getVoucherTypes • rechargePrepaidAccount • refundPrepaidVoucher • updateEmailNotification • ATS Account • Telephony Plan • Telephony Crew Plan • Telephony Prepaid Plan • Telephony Bulk Details • Telephony Bulk Plan Update • Telephony Bulk Plan • Capcode • PrepaidAccountDetail • prepaidVoucher • servicePlanOptions • TwoStage • Removed following sections: • refundStatusEnum • getEVoucherBundles • getEVoucherRechargeBundles • getMarketingCodes • getPagingBundles • getPrepaidAccountReport • getSMSBundles	Product Management

BETA

Date	Version	**Description**	Author
		• getVoiceDataComboBundles • rechargeAccount • refundVoucher • searchVoucherOrder • telephonyDevice • Other significant updates • Removed Marketing Codes from all calls • Remove CIBER extract format support • Removed scheduled transitions • Added authentication for getSystemStatus • Added commitment date ranges for OpenPort, Telephony, Paging accounts.	
06/21/2013	2.0.7	• Update SBD Destination Search and Replace to have 250 result limit to avoid timeout errors. • Update getAccountPoolingGroup report to have 500 result limit to avoid timeout errors.	Product Management
05/22/2013	2.0.6	• Add status to getAccountPoolingGroup Response as well as getAccountPoolingGroups request and response. • Add M2M device type to deliveryMethodTypeEnum	Product Management
04/25/2013	2.0.5	• Removed final Beta tags • Added validateState option to validateDeviceString to differentiate between validation for activations and validation for number correctness. • Add M2M_IMEI to subscriberAccountSearchTypeEnum.	Product Management
04/16/2013	2.0.4	• Replace original calls with beta calls, remove beta calls. • Change sbdDeliveryDetails and sbdDeliveryDestination to DeliveryDetails and deliveryDestination respectively so the objects can be shared with M2M.	Product Management
04/01/2013	2.0.3	• Added getM2MBundles • Removed getEligibleAccountsForPoolGroup • Add rudicsGroupId to m2mPlanImpl • Add MSISDN-C to M2M account object	Product Management
03/11/2013	2.0.2	• Added bulkAccountUpdate call for SBD and M2M+ bulk transitions.	Product Management
03/04/2013	2.0.1	• Correct deviceId datatype to string from long. • Removed OpenPort Bulk Details – bulk actions will not be supported for OpenPort at this time.	Product Management

BETA

Date	Version	**Description**	Author
		• Moved publicIp element from within deviceIpImpl object, to part of OpenPortSubscriberAccountImpl object.	
2/15/2013	2.0.0	• Combine Guides, add new features for Spring 2013 Release – Draft	Product Management
12/26/2012	1.3	• Updated resource links	Product Management
12/12/2012	1.2	• Added VCS bundle elements and getVCSBundles call added with Fall 2012 release.	Product Management
07/28/2011	1.1	• Final	Product Management

BETA

2 Introduction

2.1 Welcome to Iridium Web Services

The Iridium Web Services API (IWS) is a robust interface that allows you to automate provisioning processes by integrating Iridium's billing and provisioning platforms into your core business operations. IWS is based on Java web services that use Simple Object Access Protocol (SOAP) / Extensible Markup Language (XML) requests and responses. This framework will enable you to create powerful, flexible solutions. Partner applications can be developed using JAVA, C++, Microsoft .Net or other programming languages.

2.2 Iridium Web Services Connection Endpoints

There are multiple endpoints available for connecting to Iridium Web services. Initial development and testing should be performed on the training site. When Iridium is preparing to release new functionality, we will often open up the Beta environment for developers to try out the new features. Both training and beta require different credentials than production, to ensure the safety of the environments and prevent errors by connecting to the wrong place.

Once a partner has been certified in the training environment they are provided with production credentials.

2.2.1 Beta

The endpoint address for beta environment (which is only available prior to new releases)
https://iwsbeta.iridium.com:8443/iws-current/iws

2.2.2 Training (SITEST)

The endpoint address for the training environment, this is available at all times for testing current production release. This environment is usually refreshed from production data shortly after each production release to ensure current data and latest code. https://iwstraining.iridium.com:8443/iws-current/iws

2.2.3 Production

The endpoint address for production environment

https://iws.iridium.com:8443/iws-current/iws

2.3 New to Web Services?

There are many ways to learn about the technologies involved in building a web service client. A developer integrating with IWS should have a working understanding of object-oriented design, XML (and XML schemas), WSDL, and SOAP.

Depending on the language the web service client is written in, there are many tools that aid in the process, including:

- Java: Axis (wsdl2java), and JAX-WS (wsgen)
- C# and VB.NET: (wsdl.exe)
- C++: Axis (WSDL2Ws), gsoap (wsdl2h & soapcpp2)
- PHP: PHP Native (PHP-SOAP), PEAR::SOAP, NuSOAP

- Perl: SOAP::Lite

2.3.1 Iridium for Partners Provisioning Resources

All presentations, training sessions, guides, contracts, forms, etc. can be found by searching for IWS on the Iridium for Partners website.

2.3.2 Developer Documentation (Iridium)

- Iridium WSDL Explorer (https://training.iridium.com/iwsexplorer/)
    - Select the appropriate endpoint on the top right of the page for the environment you wish to connect to:
        - Beta - External
        - SITEST - External
    - The WSDL itself is accessible if you click on the Iridium Web Services logo at the top of the page.

2.3.3 Public Documentation

There are a number of Web Services resources available on the Internet. For example, the World Wide Web Consortium (W3C) and Microsoft both provide information on Web Services and XML.

3 Iridium Web Services

3.1 Architecture

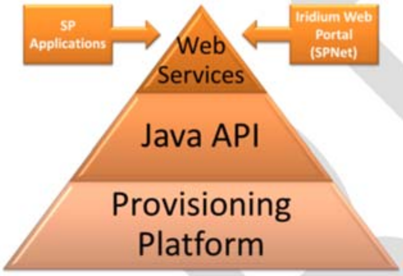


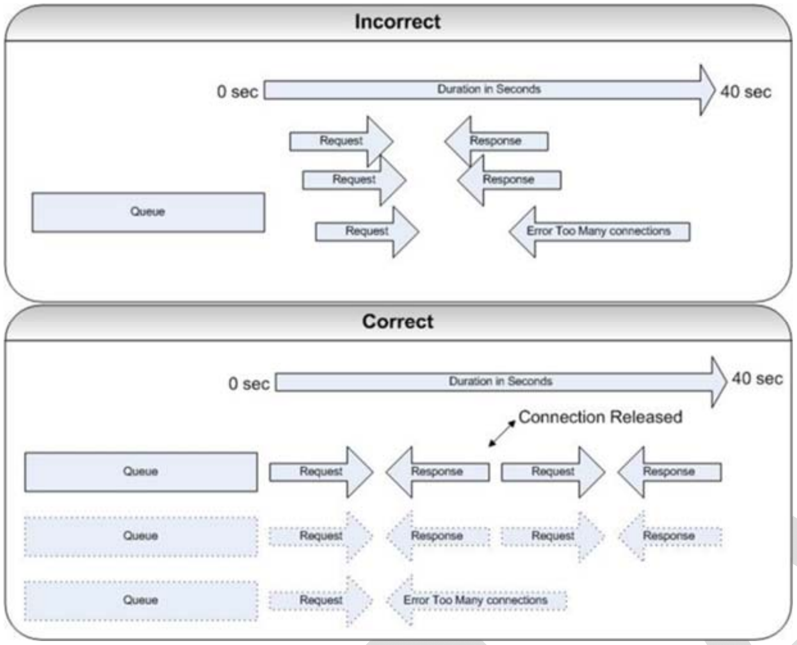
Figure 1 Basic Architecture of the IWS platform

At the core of the service is the provisioning application which is maintained and managed by Iridium. The next layer on top of the core is the Java API. It is in this layer that business intelligence is developed to work with the provisioning capabilities of the platform. On top of the Java API layer is the web service layer. This layer is how Iridium extends the features and capabilities of the provisioning platform to our partners. It is in this layer that you, our partner, will integrate with IWS.

3.2 IWS Connection Pools (all environments)

Each IWS User has a connection pool that is limited to two simultaneous requests or threads. Exceeding this limit will result in a “Too many connections!” error. Best practice is for the application to wait for the response before sending the next request on the same queue or thread.

BETA



4 Activations

4.1 What are activations?

Activations are the mechanism by which devices are associated to an account and provisioned on the network. The provisioning step makes the devices available to transmit calls and data on the Iridium network.

If a developer wants to build his/her own function to activate a subscriber, the following sections should aid in that process.

4.2 How to activate an OpenPort account

4.2.1 Step by step process

1. The first step in activating an OpenPort account is to make plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
   a.) *Demo and Trial Bundle*
   Request Get Demo and Trial Bundles and pass in your (originally empty) plan object.
   b.) *Promo Bundle*
   Request Get Promo Bundles and pass in your updated plan object that has your selected demo and trial bundle specified.
   c.) Voice and Data, VCS Bundle
      i. Request Get Voice Bundles and pass in your updated plan object that has your selections (demo and trial and promo) specified.
      ii. Request Get Data Bundles and pass in your updated plan object that has your selections (demo and trial and promo) specified.

BETA

        iii. Request Get VCS Bundles and pass in your updated plan object that has your selections (demo and trial and promo) specified.
    d.) Data Speed, Commitment Period, and Language
        i. Request Get Data Speeds and pass in your updated plan object that has your selections (demo and trial, promo, voice, and data) specified.
        ii. Request Get Commitment Periods and pass in your updated plan object that has your selections (demo and trial, promo, voice, and data) specified.
        iii. Request Get Languages and pass in your updated plan object that has your selections (demo and trial, promo, voice, and data) specified.
    e.) *Payment Model and SIM Lock*

        These are features that can be toggled or populated once the rest of your pricing data is set.

2. Next, the SIM should be specified on the account. A SIM is a number that represents the physical device sending and receiving OpenPort data.

   The SIM can be validated prior to requesting activateSubscriber by using Validate Device String.

3. Next, the IMEI can be specified on the account. The IMEI (International Mobile Equipment Identity) is only used for information purposes for OpenPort; however it is mandatory for features like SIM Lock. It is a good idea to include it if available.

4. Next, the IP and/or VPN can be specified on the account. These are both optional fields; however, they can be specified on the request.
    a. To specify these options, a list of available VPNs (as well as the IPs on each of those VPNs) can be retrieved using Get Devices.
    b. If these are not specified, IWS will utilize the default (No VPN) and auto-select an IP address that belongs to the service provider.

5. Finally, at least 1 phone line is required to be added to the account (however, up to 3 are supported). See Number for details about what this is.

6. Once the request is built, call activateSubscriber.

BETA

4.2.2 Successful Activation

If the request is successful, a response will be returned with the fully populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

4.3 How to activate an OpenPort account on a Shared Group

4.3.1 Step by step process

1. The first step in activating an OpenPort account is to make plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application

and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:

   a.) Demo and Trial Bundle
      i. Demo and Trial is not eligible for share groups, the DEMO_NONE bundle should be specified
   b.) Promo Bundle
      i. Promo selections are not eligible for share groups, the PROMO_NONE bundle should be specified
   c.) accountPoolingGroup – gather the account pooling group is of the share group this sim should join.
      i. Request Get Account Pooling Groups and pass in your desired selection, OPEN_PORT service type and either STATIC or DYNAMIC pool group type.
   d.) Voice, Data, Data Speed, VCS, Commitment – these attributes will be ignored since the group members will inherit the settings of the group, but since they are still required values pass in a 0.
      i. Voice bundle will be ignored with a share group account, 0 should be passed.
      ii. Data bundle will be ignored with a share group account, 0 should be passed.
      iii. VCS bundle will be ignored with a share group account, 0 should be passed.
      iv. Data Speed bundle will be ignored with a shared group account, 0 can be passed.
      v. Commitment bundle will be ignored with a shared group account, 0 can be passed.
   e.) Language Bundle
      i. Request Get Languages and pass in your updated plan object that has your selections (demo and trial, promo, voice, and data) specified.
   f.) *Payment Model, and SIM Lock*
      These are features that can be toggled or populated once the rest of your pricing data is set.

2. Next, the SIM should be specified on the account. A SIM is a number that represents the physical device sending and receiving OpenPort data.

   The SIM can be validated prior to requesting activateSubscriber by using Validate Device String.

3. Next, the IMEI can be specified on the account. The IMEI (International Mobile Equipment Identity) is only used for information purposes for OpenPort; however it is mandatory for features like SIM Lock. It is a good idea to include it if available.

4. Next, the IP and/or VPN can be specified on the account. These are both optional fields; however, they can be specified on the request.
   a. To specify these options, a list of available VPNs (as well as the IPs on each of those VPNs) can be retrieved using Get Devices.

BETA

    b. If these are not specified, IWS will utilize the default (No VPN) and auto-select an IP address that belongs to the service provider.

5. Finally, at least 1 phone line is required to be added to the account (however, up to 3 are supported). See Number for details about what this is.

6. Once the request is built, call activateSubscriber.

4.3.2 Successful Activation

If the request is successful, a response will be returned with the fully-populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

4.4 How to activate an SBD device

4.4.1 Step by step process

1. The first step in activating an SBD device is to make plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
    a. Demo and Trial Bundle
    Request Get Demo and Trial Bundles and pass in your (originally empty) plan object.
    b. Promo Bundle
    Request Get Promo Bundles and pass in your updated plan object that has your selected demo and trial bundle specified.
    c. SBD Bundle
    Request Get SBD Bundles and pass in your updated plan object that has your selections (demo and trial and promo) specified.
    d. LRIT flagstate, and Ring Alerts, These are features that can be toggled or populated once the rest of your pricing data is set and you are authorized by Iridium to provide LRIT services.

2. Next, the IMEI should be specified on the device. An IMEI (International Mobile Equipment Identity) is a number that represents the physical device sending and receiving short burst data messages.

    The IMEI can be validated prior to requesting activateSubscriber by using validateDeviceString.

3. Finally, at least 1 delivery destination is required to be added to the device (however, up to 5 are supported). See Delivery Destination for details about what this is. IWS also supports up to 10 MT Filters for each subscriber.

4. Once the request is built, request activateSubscriber.

BETA

4.4.2 Successful Activation

If the request is successful, a response will be returned with the fully-populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

4.5 How to activate an SBD device on a Shared Group

4.5.1 Step by step process

5. The first step in activating an SBD device is to make plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
   a. *Demo and Trial Bundle*
      i. Demo and Trial is not eligible for share groups, the DEMO_NONE bundle should be specified
   b. *Promo Bundle*
      i. Promo selections are not eligible for share groups, the PROMO_NONE bundle should be specified
   c. *accountPoolingGroup – gather the account pooling group ID of the share group this IMEI should join.*
      i. Request and pass in your desired selection, SHORT_BURST_DATA service type and either STATIC or DYNAMIC pool group type.
   d. SBD Bundle – *this attribute will be ignored since the group members will inherit the settings of the group, but since they are still required values pass in a 0.*
   e. LRIT flagstate, and Ring Alerts, These are features that can be toggled or populated once the rest of your pricing data is set and you are authorized by Iridium to provide LRIT services.

6. Next, the IMEI should be specified on the device. An IMEI (International Mobile Equipment Identity) is a number that represents the physical device sending and receiving short burst data messages.

   The IMEI can be validated prior to requesting activateSubscriber by using validateDeviceString.

7. Finally, at least 1 delivery destination is required to be added to the device (however, up to 5 are supported). See Delivery Destination for details about what this is. IWS also supports up to 10 MT Filters for each subscriber.

8. Once the request is built, request activateSubscriber.

BETA

4.5.2 Successful Activation

If the request is successful, a response will be returned with the fully-populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

4.6 How to activate a Burst device

4.6.1 Step by step process

1. The first step in activating a Burst device is to make plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
    a. Demo and Trial Bundle
    Request Get Demo and Trial Bundles and pass in your (originally empty) plan object.
    b. Promo Bundle
    Request Get Promo Bundles and pass in your updated plan object that has your selected demo and trial bundle specified.
    c. Burst Bundle
    Request Get Burst Device Bundles and pass in your updated plan object that has your selections (demo and trial and promo) specified.

2. Next, the IMEI should be specified on the device. An IMEI (International Mobile Equipment Identity) is a number that represents the physical device sending and receiving short burst data messages.

    The IMEI can be validated prior to requesting activateSubscriber by using validateDeviceString.

3. Once the request is built, request activateSubscriber.

5 Successful Activation

If the request is successful, a response will be returned with the fully-populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

5.1 How to activate an M2M+ device

5.1.1 Step by step process

1. The first step in activating an M2M+ device is to make plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
    a. Demo and Trial Bundle
    Request Get Demo and Trial Bundles and pass in your (originally empty) plan object.

BETA

    b. Promo Bundle
    Request Get Promo Bundles and pass in your updated plan object that has your selected demo and trial bundle specified.
    c. M2M+ Bundle
    Request Get M2M+ Bundles and pass in your updated plan object that has your selections (demo and trial and promo) specified.
    d. LRIT flagstate, Ring Alerts, RUDICS Group and Marketing code
    These are features that can be toggled or populated once the rest of your pricing data is set and you are authorized by Iridium to provide LRIT services.

2. Next, the IMEI should be specified on the device. An IMEI (International Mobile Equipment Identity) is a number that represents the physical device sending and receiving short burst data messages.

   The IMEI can be validated prior to requesting activateSubscriber by using validateDeviceString.

3. Finally, at least 1 delivery destination is required to be added to the device (however, up to 5 are supported). See Delivery Destination for details about what this is. IWS also supports up to 10 MT Filters for each subscriber.

4. Once the request is built, request activateSubscriber.

5.1.2 Successful Activation

If the request is successful, a response will be returned with the fully-populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

5.2 How to activate a Telephony device

5.2.1 Step by step process

1. The first step in activating a telephony device is to make service plan selections to populate the account object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
    a. Submarket
    Use validateDeviceString passing in the SIM ICCID to be activated. The results of this request will provide the submarket associated to the SIM, as well as validation that the SIM is available to the SP for activation.
    b. Service Plans
    Request Get Service Plans and pass in your updated plan object that has your selections (demo and trial, promo, commitment, submarket) specified.
    c. Service Options
    Based on service plan chosen, request the service options associated to the plan

BETA

using getServicePlan and pass in your service plan selection. Some service options will require additional information such as plus one number or RUDICS group.

   i. getServicePlan Returns the plan details and an array of options available to the plan. Included in this array are informational values to help determine what is required by the plan.
   ii. requiredGroups is an array of details for group selections used to identify option groups where the choices may be mutually exclusive, and have one or more selections required for purchase. Example: Group ”Voicemail Option” will be used for the No Voicemail, Follow Me Paging and Voicemail bundles. On any given telephony account (if multiple are available) one of these choices will be required, but only one
   iii. serviceOptions is an array of bundles available to the plan

d. Prepaid Account
If your servicePlan results indicate this is a prepaid plan (paymentModel), you may choose to activate the SIM on the prepaid platform at time of activation. Populate the prepaid plan object with the desired activation settings, and initial voucher purchase options. If activatePrepaid value is set to TRUE during an activation request, the prepaidPlan object will need to be provided. If set to False, any values specified in prepaidPlan will be ignored.

e. Crew Calling Account
If your servicePlan results indicate this is a crew plan (paymentModel), you may choose to activate the SIM on the crew platform at time of activation. Populate the crew plan object with the desired activation settings. If activateCrew value is set to TRUE during an activation request, the crewPlan object will need to be provided. If set to False, any values specified in crewPlan will be ignored.

2. Next, the IMEI can be specified on the device. An IMEI (International Mobile Equipment Identity) is a number that represents the physical device , this is required if choosing and LBS plan. The IMEI can be validated prior to requesting activateSubscriber by using validateDeviceString.

3. If desired a MSISDN and MSISDNC can be supplied. If not supplied they will be auto-generated. If a plan does not have data services, the MSISDN-C will not be assigned. The MSISDN will always be assigned to an account (even for data only packages, to enable the possibility of a plan transition at a later time.) The MSISDN(s) can be validated prior to requesting activateSubscriber by using validateDeviceString.

4. Language for voicemail and user prompts needs to be selected.

5. Complete the remaining selections for the account object as desired

6. Once the request is built, submit activateSubscriber.

5.2.2 Successful Activation

If the request is successful, a response will be returned with the fully populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and status of the account.

Initial response for activations that include Crew or Prepaid details will not return the Crew or Prepaid details. These will be available once the account has been fully processed into ACTIVE state.

5.3 How to activate a Paging device

5.3.1 Step by step process

1. The first step in activating a paging device is to make service plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
    a. Submarket
    Use validateDeviceString passing in the Capcode to be activated. The results of this request will provide the submarket associated to the capcode, as well as validation that the Capcode is available to the SP for activation.
    b. Demo and Trial Bundle
    Request Get Demo and Trial Bundles and pass in your (originally empty) plan object.
    c. Promo Bundle
    Request Get Promo Bundles and pass in your updated plan object that has your selected demo and trial bundle specified.
    d. Commitment Period
    Request Get Commitment Periods and pass in your updated plan object that has your selections (demo and trial, promo, language) specified.
    e. Service Plans
    Request Get Service Plans and pass in your updated plan object that has your selections (demo and trial, promo, commitment, submarket) specified.
    f. Service Options
    Based on service plan chosen, request the service options associated to the plan using getServicePlan and pass in your service plan selection. Some service options will require additional information such as follow me paging needs a MSISDN link.
        i. getServicePlan Returns the plan details and an array of options available to the plan. Included in this array are informational values to help determine what is required by the plan.
        ii. requiredGroups is an array of details for group selections used to identify option groups where the choices may be mutually exclusive, and have one or more selections required for purchase. Example: Group UCT_VM_FMP will be used for the Uniform Call Treatment, Follow Me Paging and Voicemail bundles. On any given telephony account (if multiple are available) one of these choices will be required, but only one
        iii. serviceOptions is an array of bundles available to the plan

2. If desired, an ISDN-A can be supplied. If not supplied they will be auto-generated. The ISDN-A can be validated prior to requesting activateSubscriber by using validateDeviceString.

3. Complete the remaining selections for the account object as required (MTD, MDAs, spReference)

4. Once the request is built, request activateSubscriber.

5.3.2 Successful Activation

If the request is successful, a response will be returned with the fully populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

5.4 How to activate a Push-to-Talk Talkgroup

5.4.1 Step by step process

1. The first step in activating a Push-to-Talk Talkgroup is to make service plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
    a. Submarket
    Use getSubmarkets request to gather the available submarket for PTT Talkgroups.
    b. Service Plans
    Request Get Service Plans and pass in your updated plan object that has your selections (demo and trial, commitment, submarket) specified.
    c. Service Options
    Based on service plan chosen, request the service options associated to the plan using getServicePlan and pass in your service plan selection.
        i. getServicePlan Returns the plan details and an array of options available to the plan. Included in this array are informational values to help determine what is required by the plan.
        ii. serviceOptions is an array of bundles available to the plan
2. Provide the Subscriber Organization ID to be linked with the new Talkgroup. These values can be obtained by issuing a getSubscriberOrg request.

3. Complete the remaining selections for the account object as required (provider reference). Note: The name value for the Talkgroup must be set from the PTT Controller interface. Once populated, it will be displayed as read-only value for integrators as well as SPNet.

4. Once the request is built, request activateSubscriber.

BETA

5.4.2 Successful Activation

If the request is successful, a response will be returned with the account number that was assigned to the talkgroup, the generated fields such as the created date and current status of the account. Depending on system response times, the account status may initially be PENDING. Once the account has been successfully provisioned through all systems, the status will change to ACTIVE and the Talkgroup ID will be assigned.

5.5 How to activate a Push-To-Talk Device

5.5.1 Step by step process

5. The first step in activating a push-to-talk device is to make service plan selections (See Plan) object. This is accomplished through a series of interactive requests between a client application and IWS (Ajax is a good way to accomplish this in a web application). There is a hierarchy for the pieces of data on the plan object; thus, the following process is recommended:
    a. Submarket
       Use getSubmarkets request to gather the available submarket for PTT device.
    b. Service Plans
       Request Get Service Plans and pass in your updated plan object that has your selections (demo and trial, commitment, submarket) specified.
    c. Service Options
       Based on service plan chosen, request the service options associated to the plan using getServicePlan and pass in your service plan selection.
        i. getServicePlan Returns the plan details and an array of options available to the plan. Included in this array are informational values to help determine what is required by the plan.
        ii. serviceOptions is an array of bundles available to the plan
6. IMEI is required, Use validateDeviceString passing in the IMEI to be activated to verify a correct number.
7. Complete the remaining selections for the account object as required (spReference)
8. Once the request is built, request activateSubscriber.

5.5.2 Successful Activation

If the request is successful, a response will be returned with the fully populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

6 Updates

6.1 What are updates?

Updates are used to modify an existing subscriber. Any of the fields on the Account object can be changed using the accountUpdate request. Some of those changes may result in fees or provisioning tasks.

If a developer wants to build his/her own function to update a subscriber, the following sections should aid in that process.

6.2 How to update an OpenPort account

6.2.1 Step by step process

The accountUpdate request consists of the following pieces of data.

1. The client should query for the account using getSubscriberAccount.
2. Once the current account object is retrieved, the user can update any or all fields on the account before submitting that object to the accountUpdate request.

NOTE: The updated account object will be validated by IWS prior to performing any updates. It is important to follow the same steps and requirements outlined in Activations when performing an update. Once the account has been validated, IWS will check each of the fields in the request against what is currently stored in the system. If a field does not match, IWS will update that field.

6.2.2 Successful Update

If the request is successful, a response will be returned with the fully-updated account on it.

6.3 How to update an SBD device

6.3.1 Step by step process

1. The client should query for the account using Get Subscriber Account.
2. Once the current account object is retrieved, the user can update any or all fields on the account before submitting that object to the accountUpdate request.

NOTE: The updated account object will be validated by IWS prior to performing any updates. It is important to follow the same steps and requirements outlined in Activations when performing an update. Once the account has been validated, IWS will check each of the fields in the request against what is currently stored in the system. If a field does not match, IWS will update that field.

6.3.2 Successful Update

If the request is successful, a response will be returned with the fully-updated account on it.

7 Deactivations and Suspensions

7.1 What is a deactivation?

Deactivations release the resources associated with a subscriber account.

For all SIM-based services (e.g., OpenPort and all Pre and Postpaid Telephony-based services) the SIM is considered disposable. When the contract is deactivated, the SIM is not recycled. If it is imperative that the SIM be used again, there is a fee to reallocate it back into the system. A request for SIM reallocation should be made to your Account Manager. OpenPort IP and all MSISDN resources are released after a 60-day quarantine period. Once released, these resources are available for association with another account.

When an SBD, Burst, or PTT account is deactivated, the IMEI is released and is immediately available for reactivation with the same or a different service provider.

Deactivations can be accomplished either through a request to setSubscriberAccountStatus (Recommended) or accountUpdate.

BETA

7.2 What is a suspension?

Suspensions turn off the service for a subscriber without actually releasing the resources (e.g., SIM, IP, NUM, IMEI, etc.) associated with the account. When an account is suspended, the device will no longer be able to send and receive messages on the network. Depending on the associated plan, the device may continue to accrue monthly service fees while in the suspended state.

Suspensions can be accomplished either through a request to setSubscriberAccountStatus (Recommended) or accountUpdate. An account in a suspended state may be re-activated at any time by a similar request to setSubscriberAccountStatus.

8 Searches and Reports

8.1 What are searches and reports?

Searches and reports are useful tools to track the status and activity of an account. Reports can extract all the information about any account and are helpful in troubleshooting any device failures. The available types of reports are:

a.) Account Search
b.) Get Subscriber Account
c.) Account Transition Report (OpenPort Only)
d.) Real Time Call Data Report (OpenPort Only)
e.) SBD Device Usage Report (SBD Only)
f.) SBD IMEI Search (SBD Only)
g.) Prepaid Account Summary (Telephony Prepaid Only)
h.) Prepaid Call History (Telephony Prepaid Only)
i.) Recharge History (Telephony Prepaid Only)
j.) Subscriber Base Report
k.) Provisioning History Report

8.1.1 Account Search

The account search can look up one or more accounts based on various search criteria. It supports wildcard (* or %) searches as well as status filters. A successful search will return a list of results that the service provider specified has access to. The list will be empty if there are no devices matching the criteria.

8.1.2 Get Subscriber Account

The Get Subscriber Account service allows a user to retrieve all of the details about a particular subscriber account. This includes the plan information, delivery destinations, MT filters, etc. A successful search will return a fully-populated subscriber account object that matches the account number the user searched for.

8.1.3 IWS Report

The IWS Report ear is a separate endpoint used specifically for reporting. Scheduled and on demand reports can be generated and delivered directly or via email. Look for more information in the IRDM Soap guide for IWS Report.

BETA

9 SBD-Specific services

SBD has some unique capabilities that aid in troubleshooting and data cleanup/maintenance. They are listed below:

a.) Send Message to Device
b.) Delete Pending Downloads
c.) SBD Destination Search and Replace

10 OpenPort Scratch Card services

Scratch cards are PIN based prepaid cards used to make voice calls the Iridium network. Scratch card services are web service calls used to find and manage scratch card orders and usage against scratch cards.

a.) Search Voucher Order
b.) Activate Scratch Cards
c.) Get Scratch Card Activation Status
d.) Scratch Card Device Usage Report

10.1.1 Search Voucher Order

This call is used to find a particular scratch card or E-voucher order entered into the Iridium system based on search criteria passed. Searches can be by purchase order number or serial number. If the request is successful, a response will be returned with details for the order found.

10.1.2 Activate Scratch Cards

This call is used to activate a particular range of scratch cards or E-vouchers. A successful request will return the serial numbers of activated cards.

10.1.3 Get Scratch Card Activation Status

This call is to search for the activation status of an order by PO or request ID. A successful request will respond with activationStatus elements.

10.1.4 Scratch Card Device Usage Report

Service that queries for and displays calls made on a particular scratch card.

11 Telephony Prepaid services

Telephony prepaid integration enables the use of eVouchers, Crew calling account and Scratchcard services for telephony sims. Along with this integration certain services are available to enable full usage of the accounts:

a.) Recharge Account
b.) Refund Voucher
c.) Activate Scratchcards
d.) Delete Prepaid Account
e.) Deactivate Scratchcards
f.) Prepaid Call History Report

BETA

11.1.1 Recharge Account

This service provides the ability to add eVouchers to a prepaid account. Depending on the plan associated to the SIM, different voucher denominations and time extensions will be available.

11.1.2 Refund Voucher

This service provides the ability to refund a voucher purchased on a prepaid account. Vouchers that have been used, or were purchased more than 90 days prior are not eligible for refund. Note, processing fees may apply for refund requests.

11.1.3 Activate ScratchCards

This service will enable a provider to activate scratchcards from inventory. Single cards, or ranges of cards may be activated in a single request.

11.1.4 Delete Prepaid Account

In some scenarios it is necessary to delete the prepaid or crew calling account from the prepaid system. Usually this is necessary when it is desired to change service plans, to enable the use of different voucher types. This action will not deactivate associated sim from the network. Note: when this action is invoked the account will lose any associated time or balance.

11.1.5 Deactivate ScratchCards

In some scenarios it is necessary to deactivate a scratchcard from the prepaid system. This action will not deactivate any sim from the network.

Note: when this action is invoked the scratchcard balance will be lost.

11.1.6 Prepaid Call history Report

This service enables the user to view a detail call history for prepaid and crew calling accounts. Providers may invoke this request to view history by account, or scratchcard serial number. Associated balance impacts will be shown for prepaid accounts and scratchcard calls.

12 Major Components

12.1 Authentication

Iridium Web Services uses its own built-in authentication scheme instead of using SOAPs header for basic or digest authentication. There are two levels of authentication. The first level of authentication allows your application to access and use all IWS web services. You will receive an Iridium Web Services username and secret key after submitting an IWS Test System Access Request form (which can be found in the Iridium for Partners Provisioning Toolkit) and accepting the Iridium Provisioning System Terms and Conditions of Use. These credentials are required to authenticate and access IWS. The table below outlines the fields that are common to every request in IWS:

Element/Field	Required?	Description
iwsUsername (type:string)	Yes	Username provided to you by Iridium for authentication against IWS.

BETA

Element/Field	Required?	Description
signature (type:string)	Yes	Signature used when authenticating against IWS. It is generated using HMAC-SHA1 encryption. * See details below about how to generate this value.
serviceProviderAccountNumber (type:string)	No	The account number associated with the service provider account.
timestamp (type:dateTime)	Yes	This is the time the request was sent to IWS. This is also the same timestamp you will use when generating the signature for the request. All timestamps in IWS follow the ISO 8601 format.
caller (type:string)	No	This should be the login for the individual making requests into IWS. This will accept any string for auditing purposes. However, if the SPNet user-related services have been used to setup your permissions, this should be the login for that user.
callerPassword (type:string)	No	This is the password for the above user. If an SPNet user ID is provided as the caller, the SPNet password can be supplied as well. If both values are provided the pair will be authenticated for a correct match of user and password. If the password is not provided, then the caller value will be used for audit purposes only, and no authentication validation against caller will be performed.

BETA

* The signature is generated using the following steps:

Concatenate the values of the action and timestamp (in that order). The action is the web service operation name; for example, “authenticatedPing”.

Calculate the HMAC-SHA1 and encode Step #1 using the Iridium-provided secret key as the key.

Convert the resulting value from Step #2 to Base64 encoded resulting in the signature used in the request.

You can test your signature and request by requesting the IWS web service request authenticatedPing.

An optional, secondary level authentication is available if you choose to create a role-based architecture when using IWS services. Passing caller and callerPassword credentials will enforce role-based authentication throughout IWS.   The validation will authenticate with the defined SPNet users and groups the users are mapped to.

If you choose to invoke roles-based access to the system, your requests to IWS must contain the primary and secondary authentication fields. The caller, callerPassword and access role directly correlate to SPNet Pro login access and, for simplicity, can be set up directly on SPNet Pro in the Manage Users section. For information on managing users on SPNet Pro, please reference the Iridium SPNet Pro User Guide.

12.2 DIDWW Authentication

This is a special authentication object that extends off the IWS authentication. This includes an additional field used to submit a API key to access Iridium custom calls to manage local number resources in both Iridium and the DIDWW vendor. This provides access to local number of various countries found in the getLocalNumberRegions IWS call. The table below outlines the extended authentication object didwwAuthenticatedRequest.

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication. For this request calling user and password are required.
didwwUserApiKey (type: string)	Yes	API key provided by DIDWW vendor. Used as authentication for Iridium calls to make updates on behalf of the requestor.

12.3 Bundles and Service Plans

Bundles represent pricing options. There are 4 types of bundles. Each of those types is outlined below. Telephony and Paging services will utilize service plans and service options to select and enable pricing and features on an account, in addition to bundle selections.

12.3.1 Bundle

Most bundles in the system are this type. (bundleInfoImpl)

BETA

Element/Field	Required?	Description
Bundle (type:bundleInfoImpl)	Yes	
id(type:string)	Yes	This is the ID that the system uses to identify the bundle. It is the value that will be passed into the Plan objects.
name (type:string)	Yes	This is the display name of the bundle. It is the user-friendly **Description** for use in drop down boxes.
obsolete (type:boolean)	Yes	true false This flag indicates whether the bundle is still being offered or not. Typically it will be false. If this is true; however, it means that pricing is being discontinued and the current account has been grandfathered. If the user wants to update any other Plan options at this time, they will be forced to move off of the grandfathered bundles. The user can stay on their current Plan options and update other account data, however.

12.3.2 Bulk Bundle

There are a few bundles in the system of this type. It is used when a bundle represents a bunch (or pack) of items being purchased.

This type of bundle is utilized by some of the scratch card services for OpenPort.

Element/Field	Required?	Description
bulkBundle (type:bulkBundleInfoImpl)	Yes	
Bundle Data (type:bundleInfoImpl)	Yes	See *Bundle*.
packQuantity (type:int)	Yes	The number of items included in this bundle.

BETA

12.3.3 Demo and Trial Bundle

These bundles represent demo and trial options available. These are typically limited to a certain number per Service Provider.

Element/Field	Required?	Description
datBundle (type: datBundleInfoImpl)	Yes	
Bundle Data (type:bundleInfoImpl)	Yes	See *Bundle*.
numAvailable (type:int)	Yes	The number of these bundles remaining based on the configurable allowance.
datType (type:datTypeEnum)	Yes	NA – This is what the type will be on the "None" option. DEMO – These are demo/test bundles. They do not have an expiration date and typically have very low or no fees associated with them. TRIAL – These are trial bundles. They have an expiration date, usually 1-3 months depending on the trial. During the trial period they have very low or no fees associated with them.

12.3.4 Account Pooling Bundle

These bundles represent account pooling bundles available. These will be custom per Service Provider.

Element/Field	Required?	Description
accountPoolingBundle (type: accountPoolingBundleInfoImpl)	Yes	
Bundle Data (type:bundleInfoImpl)	Yes	See *Bundle*.

BETA

12.3.5 Service Plan

Service Plan is utilized by Telephony and Paging accounts. The new pricing model will allow for simpler addition and selection of value added services.

The service plan object will contain the following elements:

Element/Field	**Required?**	**Description**
servicePlan (type: servicePlanImpl)	Yes	Service plan object
servicePlan (type: absServicePlanImpl)	Yes	
id (type: string)	Yes	Internal system identifier for the service plan. This is to be used when adding a service plan to an activation or update request.
name (type: string)	Yes	Name of the service plan
servicePlanType (type: string)	Yes	Type of plan returned: Shared: plan returned is a shared group ID (Account pooling group) Individual: plan returned is an individual plan type
obsolete (type: boolean)	Yes	true false This flag indicates whether the plan is still being offered or not. Typically it will be false. If this is true; however, it means that pricing is being discontinued and the current account has been grandfathered. If the user wants to update any other Plan options at this time, they will be forced to move off of the grandfathered service plan. The user can stay on their current Plan options and update other account data, however.
demoTrial (type: datTypeEnum)	No	Indicates if the plan is of demo or trial type. See DAT types
prepaidPlan (type: string)	No	The prepaid/crew plan ID linked to the plan for scratchcards and e-vouchers.

BETA

Element/Field	Required?	Description
serviceCategories	No	An array of service categories in which the plan can be used.
serviceCategory (type: modelServiceCategoryEnum)	No	See Model Category
permittedServices	No	An array of options permitted for use with this plan
permittedService (type: permittedServiceImpl)	No	See Sub-system or available Features
required (type: Boolean)	No	TRUE - Feature is required FALSE - Feature is optional
requiredGroups	No	An array of group settings associated to the plan.
requiredGroup (type: requiredGroupImpl)	No	
name (type:string)	Yes	Service Option group name
requiredNum (type:int)	Yes	The number of selections required within this group.
optionalGroups	No	An array of group settings associated to the plan.
optionalGroup (type: optionalGroupImpl)	No	
name (type:string)	Yes	Service Option group name
maxNum (type:int)	Yes	The maximum number of selections permitted within this group.
servicePlanOptions	No	This will be an array of service options available with this plan
servicePlanOption (type: servicePlanOptionImpl)	No	This is individual planOption available. This could be one to many. See Service Plan Options.
twoStageBlock (type:Boolean)	No	true false This flag indicates whether the plan has access to two stage options. If FALSE,

Element/Field	Required?	Description
		two stage settings on activations or updates are ignored for processing.
optionalBundleGroups	No	An array of standalone bundle group settings associated to the plan.
optionalBundleGroup (type: optionalGroupImpl)	No	
name (type:string)	Yes	Service Option group name
maxNum (type:int)	Yes	The maximum number of selections permitted within this group.
bundleOptions	No	This will be an array of standalone bundle options available with this plan
bundleOption (type: bundleOptionImpl)	No	This is individual planOption available. This could be one to many. See Bundle Option.

12.3.6 Service Plan Options

This is the object that represents the selection of services to be included in the plan. Each service plan will have various options associated to the plan setup. Some will be required, and others optional selections. Those that are required will have Required object set to TRUE, these must be included in activate and update requests. Those with Required set to FALSE are optional services and may be selected or ignored on activation and update requests. (Note that some optional services may result in additional fees for the account, see your pricing guide to determine which apply.)

Element/Field	Required?	Description
servicePlanOption (type: servicePlanOptionImpl)	Yes	Service plan option object
id (type: string)	Yes	Internal system identifier for the service plan option. This is to be used when adding options to an activation or update
name (type: string)	Yes	This is the name of the option that is available in the plan.
required (type: Boolean)	Yes	True False This states whether this option is mandatory for the service plan. When

BETA

Element/Field	Required?	Description
		passing the ID back if this option is required and not passed on activation IWS will add it automatically.
defaulted (type: Boolean)	Yes	True False This will be the default selected option for the service plan options, utilized by SPNet, but can also be used by external systems if desired.
displayColumn (type: string)	Yes	
displayOrder (type: string)	Yes	Ordering of the service options, utilized by SPNet, but can also be used by external systems if desired. Formatted as Column / Order in column. Example: 1/3 is column 1 position 3.
groupName (type: string)	No	Some service options are by their nature mutually exclusive. To accommodate this feature some of the options will be grouped by groupName. This value in combination with the requiredGroup object at the plan level will indicate how many of the options in a group must be selected. In most cases the selection will require one in a group, and no more than one.
requires (type: string)	No	Indicates additional information is required with the purchase of this deal. Example: Additional plan selection for rudicsGroup will be required with selection of RUDICS deal. For Follow-Me paging, a Follow-Me MSISDN number will need to be provided.
obsolete (type: boolean)	Yes	true false This flag indicates whether the service plan option is still being offered or not.
dependentDeal (type: string)	Yes	

BETA

12.3.7 Bundle Option

This is the object that represents the selection of standalone bundles that can be included in a Certus plan. Each service plan will have various bundle options associated to the plan setup. Each bundle option will belong to a optional bundle group. Those that are required will have Required object set to TRUE, these must be included in activate and update requests. Those with Required set to FALSE are optional bundles and may be selected or ignored on activation and update requests. (Note that some optional bundles may result in additional fees for the account, see your pricing guide to determine which apply.)

Element/Field	**Required?**	**Description**
bundleOption (type: bundleOptionImpl)	Yes	Bundle option object
id (type: string)	Yes	Internal system identifier for the bundle option. This is to be used when adding options to an activation or update
name (type: string)	Yes	This is the name of the option that is available in the plan.
subservice (type: string)	Yes	This indicates which Certus sub-service the bundle belongs to.
required (type: Boolean)	Yes	True False This states whether this option is mandatory for the service plan. When passing the ID back if this option is required and not passed on activation IWS will add it automatically.
displayOrder (type: string)	Yes	Ordering of the service options, utilized by SPNet, but can also be used by external systems if desired. Formatted as Column / Order in column. Example: 1/3 is column 1 position 3.
groupName (type: string)	No	All bundle options are mutually exclusive. To accommodate this feature all options will be grouped by groupName. This value in combination with the requiredGroup object at the plan level will indicate how many of the options in a group must be selected. In most cases the selection will require one in a group, and no more than one.

BETA

Element/Field	Required?	Description
requires (type: string)	No	Indicates additional information is required with the purchase of this deal.
obsolete (type: boolean)	Yes	true false This flag indicates whether the bundle option is still being offered or not.
dependentDeal (type: string)	Yes	

12.3.8 Permitted Service

The permitted service shows the sub-system available for a Certus service plan.

Element/Field	Required?	Description
permittedService (type: permittedServiceImpl)	Yes	
service (type: purchaseConditionEnum)	Yes	See Purchase Conditions
required (type: boolean)	Yes	Indicates if a service is required on activation or update.

BETA

12.4 Variable Types / Values

There are a number of type fields utilized by IWS. Below are the rules regarding types, and the valid values for custom types.

Base types:

- string
- dateTime
- boolean
- int
- long
- double
- float

12.4.1 DateTime String Format

In version 2.0, most dateTime values changed to string values. This is to allow for acceptance of null values in the input and response. The format for all date variables of string type is: yyyy-MM-dd'T'HH:mm:ss'Z'

Examples:

```
<startDate>2012-01-01T00:00:00Z</startDate>
<endDate>2014-01-01T00:00:00Z</endDate>
```

12.4.2 accountStatusEnum

Used for account status values. Values are of type String. Valid Values are:

Value	**Description**
ACTIVE	The account is active and can be used to make requests/generate usage.
DEACTIVE	The service has been deactivated. It cannot be used to make requests and it is not generating monthly charges. A device in this state cannot be re-provisioned for use through the web services.
ERROR	The account is in an error state. Contact Tier 2 for assistance to clear this condition.
INCOMPLETE	The primary service for the account is ACTIVE, but one or more of the secondary services is in an ERROR state.
PENDING	The device has outstanding provisioning service orders that haven't completed processing. It cannot be modified or used while it is in this state. NOTE: This is a derived state (i.e., determined from other pieces of data in the system).
PREPAID_ERROR	The account is valid in billing, but there is an error on the prepaid side,
REQUESTED	The device has outstanding provisioning service orders that are waiting to be submitted for processing. It cannot be modified or used while it is in this state. NOTE: This is a derived state (i.e., determined from other pieces of data in the system).
SUSPENDED	The service on the device has been suspended so that it cannot be used to make requests. While in this state, the account may or may not be generating monthly fees depending on the plan. An account in this state can have the service turned back on.

12.4.3 activationStatusEnum

Used for activation status values. Values are of type String. Valid Values are:

Value	**Description**

BETA

ERROR	Request has failed processing
FINISHED	Request has finished processing
PENDING	Request is Pending completion

12.4.4 activationTypeEnum

Used for activation type values. Values are of type String. Valid Values are:

Value	**Description**
FULL	Activate all scratch cards on the order
LINEITEM	Activate all scratch cards associated with a particular line item.
RANGE	Activate a contiguous serial range of scratch cards

12.4.5 atsChannelTypeEnum

Used for ATS account settings. Values are of type String. Valid Values are:

Value	**Description**
PRIMARY	Set as Primary account for ATS pair
SECONDARY	Set as Secondary account for ATS pair.

12.4.6 bulkPrepaidTypeEnum

Used for bulk prepaid. Values are of type String. Valid values are:

Value	**Description**
PREPAID	Designates request as prepaid
CREW	Designates request as crew

BETA

12.4.7 businessTypeEnum

Used for identifying business type values. Values are of type String. Valid Values are:

Value	Description
BURSTSERVICE	
BUSINESS	
CERTUS	
CONSUMER	
M2M	
OP	
PAGING	
PTT_TG	
SBD	
SCRATCH_CARD	
SHARING_GROUP	
TEL	
UNKNOWN	
NOT_APPLICABLE	

12.4.8 callDirectionEnum

Used for call direction values. Values are of type String. Valid Values are:

Value	Description
CALLFORWARD	The subscriber forwarded the call to another number.
EMERGENCY	The subscriber made a call to the emergency number.
INCOMING	The subscriber received the call.
OUTGOING	The subscriber made the call.

12.4.9 callStatusEnum

Used for call status values. Values are of type String. Valid Values are:

Value	Description
ENDED	Call has completed, does not indicate reason for completion.
IN_PROGRESS	Call is still in progress

DRAFT

12.4.10 callServiceCodeEnum

Used for call service code values. Values are of type String. Valid Values are:

Value	Description
ALL	
DATA	
DOWNLOAD	
ERROR	
NOTDELIVERED	
PENDING	
SMS	
STREAMING	
UPLOAD	
VOICE	

12.4.11 certusLineTypeEnum,

Used for identifying the type of voice line for Certus. Values are of type String. Valid Values are:

Value	Description
POSTPAID	Line is primarily used for postpaid traffic
PREPAID	Line is primarily used for Prepaid traffic
SAFETY	Line is provisioned to support safety services

12.4.12 certusSecondaryDataServicePriorityEnum

Used for identifying secondary data service priority. Values are of type String. Valid Values are:

Value	Description
EXPEDITED	flow with higher priority if contention exists
NORMAL	flow will share normal priority if contention exists.

12.4.13 coloDataNWOptionsEnum

Used for identifying what networks are permitted for provisioning for a model. Values are of type String. Valid Values are:

Value	Description
NONE	No data network permitted
SP_COLO	Only SP defined colo permitted on contract
IRIDIUM_COLO	Only Iridium defined colo permitted on contract
SP_VPN	Only VPN permitted on contract
IRIDIUM_COLO_SP_COLO	Allows Iridium colo and SP colo
SP_VPN_SP_COLO	Allows SP VPN and SP colo
SP_VPN_IRIDIUM_COLO	Allows SP VPN and Iridium Colo
All_COLO_DATANW_OPTIONS_ON	Allows all data networks

BETA

12.4.14 contractTypeEnum

Used for search filter on vessels. Values are of type String. Valid Values are:

Value	Description
NONE	
SAFETY	
STANDARD	
ALL	

12.4.15 dataClassmarkEnum

Used for identifying Certus Data Classmark values. Values are of type String. Valid Values are:

Value	Description
BBT1	
BBT2	
BBT3	
BBT4	
HHT1	
SLBT1	
SLBT2	
SLBT3	
SLBT4	
SLBT5	

DRAFT

12.4.16 dataPrepaidOptionsEnum

Used for identifying the type of prepaid permitted for a Certus model. Values are of type String. Valid Values are:

Value	Description
NONE	Prepaid not permitted
SCRATCHCARD	Allows scratchcard based prepaid
SIM	Allows SIM based prepaid
SCRATCHCARD_AND_SIM	Allows both scratchcard and SIM based prepaid

12.4.17 dateRangeTypeEnum

Used for identifying date range values for search and reporting actions. Values are of type String. Valid Values are:

Value	Description
ALL	No date range filter.
CUSTOM	User supplied date range values
MTD	Calculate date range as 1st of current month until now
PREVMONTH	Calculate date range as beginning to end of prior month
PREVWEEK	Calculate date range as beginning to end of previous week
TODAY	Calculate date range as beginning of today until now
WTD	Calculate date range as beginning of current week until now
YESTERDAY	Calculate date range as beginning to end of yesterday
YTD	Calculate date range as January 1 until now

12.4.18 datTypeEnum

Used for account type values. Values are of type String. Valid Values are:

Value	Description
DEMO	Demo account type – limited charges applied to this type of account
NONE	Regular commercial account
TRIAL	Trial account type – will transition to regular plan after certain period

12.4.19 deliveryMethodTypeEnum

Used for SBD service delivery destination type values. Values are of type String. Valid Values are:

Value	Description
DIRECT_IP	Delivery destination is an IP address with port.
EMAIL	Delivery destination is an email address
IRIDIUM_DEVICE	Delivery destination is an Iridium device supporting SBD traffic

BETA

12.4.20 deliveryStatusTypeEnum

Used for SBD service delivery status result type values. Values are of type String. Valid Values are:

Value	Description
DELIVERED	
DELIVERYCONFIRMED	
DIRECTIP_DELETED	
DIRECTIP_FLUSHED	
DIRECTIP_QUEUEOVERFLOW	
DIRECTIP_TIMEOUT	
INVALIDEMAIL	
NOTDELIVERED	
SSDTOSSD_EMBC	
SSDTOSSD_PAYLOADINVALID	
SSDTOSSD_QUEUEFULL	
SSDTOSSD_UNKNOWNSSD	

12.4.21 deviceNumberTypeEnum

Used for device number type values. Values are of type String. Valid Values are:

Value	Description
ISDNA	
LOCALNUMBER	
MSISDN	
MSISDNB	
MSISDNC	
PLUS1	
PLUS7	

BETA

12.4.22 deviceStatusTypeEnum

Used for resource inventory status values. Values are of type String. Valid Values are:

Value	**Description**
ACTIVE	The device has been provisioned on an account and is currently available for transmitting call data.
INACTIVE	The device was previously on an account; however, it has been un-provisioned and is no longer available for transmitting call data. Devices in this state cannot be re-used without contacting Service Delivery for assistance.
RESERVED	The device is reserved for use by a particular service provider. It can be provisioned on a new or existing account.
UNKNOWN	This is indicative of a system error. It means that IWS could not map the state of the device to one of the other types probably due to a configuration issue.

12.4.23 deviceStringTypeEnum

Used for defining types for validateDeviceString. Values are of type String. Valid Values are:

Value	**Description**
ATS_ICAO	This is used to locate by ICAO
ATS_TAIL_NUMBER	This is used to locate by Tail Number.
CAPCODE	This is used to locate paging Capcodes.
IMEI	This is used to locate IMEIs.
IPADDR	This is used to locate IP Addresses.
ISDNA	This is used to locate ISDNAs.
MSISDN	This is used to locate MSISDNs.
MSISDNB	This is used to locate MSISDNBs.
MSISDNC	This is used to locate MSISDNCs.
PLUS1	This is used to locate PLUS 1 numbers
PLUS7	This is used to locate PLUS 7 numbers
SIM	This is used to locate SIM values

DRAFT

12.4.24 deviceTypeEnum

Used for specifying values for number searches. Values are of type String. Valid Values are:

Value	Description
CAPCODE	This is to be used to locate Paging capcodes.
IMEI	This is to be used to locate IMEI values.
IP	This is to be used to locate IP addresses.
ISDNA	This is to be used to locate ISDNA numbers.
LOCAL_NUMBER	This is used to find Local numbers
NUM	This is to be used to locate MSISDN numbers.
PLUSONE	This is used to find Plus 1 numbers
SIM	This is to be used to locate SIM values.
VOUCHER	This is used to locate scratch cards.
VPN	This is to be used to locate VPN ids.

12.4.25 deviceUsageTypeEnum

Used for specifying values for device usage for a service provider. Values are of type String. Valid Values are:

Value	Description
ANY_DEVICE	pull from both SPs device pool and global
SP_ONLY	pull devices from the SPs device pool
SP_PRIORITY	pull devices from SP device pool and only pull devices from the GLOBAL pool if no devices are available in SP pool
GLOBAL_PRIORITY	pull devices from GLOBAL device pool and only pull devices from SPs pool if no devices are available in the GLOBAL pool.
GLOBAL_ONLY	pull devices from the GLOBAL device pool

12.4.26 emailNotificationTypeEnum

Used to available actions for email notifications. Values are of type String. Valid Values are:

Value	Description
ACT_CREATE	Activations
DE_ACTIVATE	Deactivations
DEV_SWAP	Device Swaps
PLAN_TRANS	Plan transitions
SUPP_OFF	Remove Services
SUPP_ON	Add services
SUSPEND	Suspensions
UN_SUSPEND	UnSuspensions

BETA

12.4.27 endpointEnum

Used to distinguish IWS endpoints. Values are of type String. Valid Values are:

Value	Description
IWS	
IWSREPORT	
IWSREPORTENGINE	

12.4.28 errorCodeEnum

Used to indicate type of error code. Values are of type String. Valid Values are:

Value	Description
AUTH	
CONFIG	
FIELD	
SYS	

12.4.29 extractionFormatEnum

Used to specify the type of data extract format. Values are of type String. Valid Values are:

Value	Description
NA	
NATAP2	
TAP2	

12.4.30 homeGatewayEnum

Used to choose the home gateway for devices which can be provisioned at multiple gateways. Values are of type String. Valid Values are:

Value	Description
GLOBAL	This device is provisioned to be homed to the Global gateway (assumes Global SIM to be in the device for traffic to route properly)
RUSSIA	This device is provisioned to be homed to the Russia gateway (assumes Russia SIM to be in the device for traffic to route properly)

12.4.31 imeiTypeEnum

Used for IMEI type values. Values are of type String. Valid Values are:

Value	Description
LBS_IMEI	
REGULAR_IMEI	

12.4.32 imeiSimLockPriorityEnum

Used to denote the SIM Lock requirements for Certus and OpenPort when provisioning a contract. Values are of type String. Valid Values are:

DRAFT

Value	Description
REQUIRED_DISABLED	SIM Lock not permitted on any provider contracts
REQUIRED_ENABLED	SIM Lock required on any proved contract
DEFAULT_DISABLED	SIM Lock permitted default enabled on SPNet
DEFAULT_ENABLED	SIM Lock permitted default disabled on SPNet

12.4.33 iwsLogTypeEnum,

Used for identifying the requested values returned for IWS logging. Values are of type String. Valid Values are:

Value	Description
REQUEST	
RESPONSE	
BOTH	

12.4.34 languageTypeEnum,

Used for identifying language type values. Values are of type String. Valid Values are:

Value	Description
ENGLISH	
FRENCH	

12.4.35 lineQualityEnum,

Used for identifying Certus voice line quality values. Values are of type String. Valid Values are:

Value	Description
HIGH	High quality voice line (2 channels)
STANDARD	Standard quality voice line (1 channel)

12.4.36 lineTypeEnum

Used for line type indicators. Values are of type String. Valid Values are:

Value	Description
CREW	
STANDARD	

12.4.37 maxDataBandwidthEnum

Used to identify the max data speed for a Certus model. Values are of type String. Valid Values are:

Value	Description
KBPS_0	
KBPS_22	
KBPS_88	
KBPS_176	
KBPS_352	
KBPS_704	

BETA

12.4.38 modelServiceCategoryEnum,

Used for identifying Certus model category values. Values are of type String. Valid Values are:

Value	Description
SFX_STANDARD	
STANDARD	
HIGH	
ULTRA	
OTHER	

12.4.39 modelServiceClassEnum,

Used for identifying Certus model class values. Values are of type String. Valid Values are:

Value	Description
CERTUS_100	Certus 100 class device
CERTUS_1500	Certus 1500 class device
CERTUS_200	Certus 200 class device
CERTUS_350	Certus 350 class device
CERTUS_700	Certus 700 class device
IHANDSET	Telephony handset device, such as 9555
ISBD	SBD class device, such as 9603
ILBT	LBT class device, such as 9523
IWIFI	Wi-Fi class device, such as GO

BETA

12.4.40 mtFilterRuleTypeEnum

Used for SBD MT filter rule types. Values are of type String. Valid Values are:

Value	Description
DOMAIN	
EMAIL	
IP_ADDRESS	

12.4.41 networkEnum,

Used for identifying network element type values. Values are of type String. Valid Values are:

Value	Description
PROD	
TEST	

12.4.42 objectStatusEnum

Used for identifying object status values. Values are of type String. Valid Values are:

Value	Description
ACTIVE	Active objects, valid for actions
OBSOLETE	Obsolete objects, no longer valid for new actions.

12.4.43 operationStatusEnum

Used for identifying operation status values. Values are of type String. Valid Values are:

Value	Description
ALT_SOURCE	Operation is reading from backup datasource
DISABLED	Operation has been disabled and is unavailable.
ENABLED	Operation is enabled for normal use
FAIL	Operation will reject all requests due to related component outages. The status will be set back to enabled once the outage has been resolved.
QUEUED	Operation is queueing requests for processing once related component outages have been resolved
REPLAY	Operation is currently processing queued requests. New requests will be added to the end of the queue.

12.4.44 outageStatusEnum

Used for identifying scheduled outage status values. Values are of type String. Valid Values are:

Value	Description
SCHEDULED	Outage is scheduled for future occurrence
WORKING	Outage is currently active
DONE	Outage has completed
CANCELLED	Outage was cancelled by administrators

12.4.45 paymentModelTypeEnum

Used for payment model. Values are of type String. Valid Values are:

Value	Description
POST_PAID	This is used to locate POST PAID.
PRE_PAID	This is used to locate PRE PAID.

12.4.46 poolGroupTypeEnum

Used for specifying pool group type. Values are of type String. Valid Values are:

Value	Description
DYNAMIC	
STATIC	

12.4.47 prepaidAccountTypeEnum

Used for prepaid account type. Values are of type String. Valid Values are:

Value	Description
CREW	This is used to locate CREW
PREPAID	This is used to locate PREPAID.
CERTUS	This is used to locate CERTUS

BETA

12.4.48 prepaidLanguageTypeEnum

Used for prepaid languages. Values are of type String. Valid Values are:

Value	Description
ENGLISH	

12.4.49 primaryDataServiceTypeEnum

Used for Certus primary data services. Values are of type String. Valid Values are:

Value	Description
POSTPAID	
PREPAID	

12.4.50 pttTypeEnum

Used for Push-To-Talk types. Values are of type String. Valid Values are:

Value	Description
ALL	Include both Talkgroups and devices
PTT_DEVICE	Only devices
PTT_TALKGROUP	Only Talkgroups

BETA

12.4.51 queueEntryStatusEnum

Used to identify the current status of queued requests. Valid Values are:

Value	**Description**
CANCELLED	Request has been cancelled by the user
DONE	Processing has been completed successfully.
ERROR	Result of processing is ERROR
PENDING	Request is waiting for status of related components to be enabled.
READY	Request is available for processing
WORKING	Request is being processed
BILLING	Request is waiting for Billing to finish

12.4.52 purchaseConditionEnum

Used for identifying items which have variable purchase condition values. Values are of type String. Valid Values are:

Value	**Description**
BURST	
LOCAL_NUMBER	
PTT	
PUBLIC_IP	
SAFETY	
SBD	
STREAMING	
VOICE	

12.4.53 rudicsTypeEnum

Used to specify the type of Rudics group being returned on a Telephony contract. Values are of type String. Valid Values are:

Value	**Description**
RUDICS	Commercial RUDICS group configured for SP
GMDSS	RUDICS group created for GMDSS safety

12.4.54 sbdActivityOrderByTypeEnum

Used for specifying the order of SBD activity values. Values are of type String. Valid Values are:

Value	**Description**
CALL_STATUS	
CEP_RADIUS	
DELIVERY_METHOD	
DELIVERY_STATUS	
LATITUDE	
LONGITUDE	
MOMSN	
UPLOAD_CALL_NUMBER	
UPLOAD_MESSAGE_SIZE	

BETA

Value	Description
UPLOAD_TIMESTAMP	

12.4.55 sbdDeleteDownloadResultTypeEnum

Used for identifying the download delete request status values. Values are of type String. Valid Values are:

Value	Description
FAILED_DID_NOT_FIND	The system was unable to locate a message with the ID provided.
FAILED_INVALID_STATE	The system found the message with the ID provided, but it was not in the PENDING state.
STATUS_UNKNOWN	The message may or may not have been deleted. Re-query for downloads to check the status.
SUCCEEDED	The message was moved to the deleted state.

12.4.56 sbdDestSearchAndReplaceResultTypeEnum

Used for identifying the search and replace result values. Values are of type String. Valid Values are:

Value	Description
DUPLICATE	This replace operation can’t/or did not take place because the same destination already exists on the account.
FAILED	The replace/delete operation can’t/or did not take place for some other reason. Probably due to system errors.
INVALID_FOR_METHOD	The replace operation can’t/or did not take place because the delivery methods do not match on the new and old values.
OK	This replace operation can/or did take place.
ONLY_DESTINATION	A delete operation can’t/or did not take place because it is the only destination on an account. At least 1 is required.
NOT_ALLOWED_FOR_RUSSIA	An update cannot be performed when changing to EMAIL type when contract is a RUSSIA account.

DRAFT

12.4.57 sbdSearchTypeEnum

Used for SBD search type values. Values are of type String. Valid Values are:

Value	Description
ACCOUNT_NUMBER	
IMEI	
IP	

12.4.58 sbdSendMessageStatusTypeEnum

Used for identifying SBD send message result status. Values are of type String. Valid Values are:

Value	Description
FAILED	
SUCCEEDED	

12.4.59 sbdUsageSearchTypeEnum

Used for specifying SBD search type values. Values are of type String. Valid Values are:

Value	Description
ALL	This will pull all MT and Mobile Originated (MO) usage that has completed. This will not pull PENDING messages.
DOWNLOAD	This will pull all MT usage that has completed.
ERROR	This will pull all MO usage that encountered errors.
NOTDELIVERED	This will pull all MO usage that hasn't yet been delivered.
PENDING	This will pull all pending MT usage.
UPLOAD	This will pull all MO usage.

12.4.60 scratchcardOrderStatusEnum

Used for scratch card orders to indicate status of order. Values are of type String. Valid Values are:

Value	Description
CREATED	Order has been created and all scratchcards are ready for activation
PROCESSING	Order has some scratchcards activated while others still await activation
COMPLETE	All scratchcards in order have been activated
ERROR	One or more cards are in error state after failed activation

12.4.61 scratchCardActivationStatusSearchTypeEnum

Used for scratch card activation status search type values. Values are of type String. Valid Values are:

Value	Description
PURCHASE_ORDER	Search for all activation requests associated with the order
REQUEST_ID	Search for a particular activation request by ID

BETA

12.4.62 scratchcardQuantityEnum

Used for denoting the requested scratchcard quantity when ordering Certus scratchcards. Values are of type String. Valid Values are:

Value	Description
QTY_50	
QTY_100	
QTY_150	
QTY_200	
QTY_250	
QTY_300	

12.4.63 scratchCardSearchTypeEnum

Used for specifying scratch card search type values. Values are of type String. Valid Values are:

Value	Description
ACCOUNT_NUMBER	(Valid for OpenPort Only) Used to search by Account Number linked to scratchcard
PIN	(Valid for OpenPort Only) Used to search by PIN Number of the scratchcard
SERIAL_NUMBER	Used to search by Serial Number of the scratchcard

12.4.64 scratchCardServiceTypeEnum

Used for specifying scratch card search type values. Values are of type String. Valid Values are:

Value	Description
OPENPORT	Filter by OpenPort
TELEPHONY	Filter by Telephony
CERTUS	Filter by Certus

BETA

12.4.65 scratchcardStatusEnum

Used to indicate if a scratchcard successfully provisioned. Values are of type String. Valid Values are:

Value	Description
SUCCESS	
FAILED	

12.4.66 searchCondEnum,

Used for identifying search action filter conditions. Values are of type String. Valid Values are:

Value	Description
CONTAINS	System will look for occurrences where the provided value is found in the search field (not case sensitive) * character is not supported
DOES_NOT_CONTAIN	System will look for occurrences where the provided value is not found in the search field (not case sensitive) * character is not supported
EQUALS	System will look for exact match to provided value (not case sensitive)
LIKE	System will look for wildcard match to provided value (not case sensitive) * character in value indicates wildcard criteria, no * provided functions same as CONTAINS search
NOT_EQUAL	System will look for all occurrences where an exact match to provided value does not exist (not case sensitive)
NOT_LIKE	System will look for all occurrences where the wildcard match to provided value is not found (not case sensitive) * character in value indicates wildcard criteria, no * provided functions same as DOES_NOT_CONTAIN search

BETA

12.4.67 searchOperatorEnum

Used for specifying search operator type values. Values are of type String. Valid Values are:

Value	Description
EQUAL_TO	
GREATER_THAN	
GREATER_THAN_EQUAL_TO	
LESS_THAN	
LESS_THAN_EQUAL_TO	
LIKE	

12.4.68 secondaryDataServiceBillingIdEnum

Used for specifying VLAN position. Values are of type String. Valid Values are:

Value	Description
SDF1_VLAN5	
SDF2_VLAN6	
SDF3_VLAN7	
SDF4_VLAN8	

12.4.69 secondaryDataServiceOverrideEnum

Used for specifying if a secondary data service on a Certus contract will use the base enabled values of the service. Values are of type String. Valid Values are:

Value	Description
DEFAULT	Assume attributes on service definition at time of session
ENABLED	Override value of SDF with enabled status
DISABLED	Override value of SDF with disabled status

12.4.70 secondarySvcStatusEnum

Used for identifying secondary service status values. Values are of type String. Valid Values are:

Value	Description
ACTIVE	Secondary service is active
ALL	
ERROR	Secondary service is in error
NOT_ENABLED	Secondary service has not been requested
PENDING	Secondary service is pending service order completion
REQUESTED	Secondary service has been requested, but is not yet provisioned to network.
SUSPENDED	Secondary service is suspended

12.4.71 secondarySvcTypeEnum

Used for identifying secondary service types used for search filters. Values are of type String. Valid Values are:

Value	Description
BURST_DEVICE	contracts which have a secondary service of Burst
CERTUS_STREAMING	contracts which have a secondary service of Certus Streaming
CERTUS_VOICE	contracts which have a secondary service of Certus Voice
PTT_DEVICE	contracts which have a secondary service of PTT
SHORT_BURST_DATA	contracts which have a secondary service of SBD

BETA

12.4.72 serviceLanguageTypeEnum

Used for identifying language prompt choices depending on service type. Values are of type String. Valid Values are:

Value	Description
CERTUS_POSTPAID	To select language choices for Certus Postpaid prompts
CERTUS_PREPAID	To select language choices for Certus Prepaid prompts
PAGING_POSTPAID	For future use
TELEPHONY_POSTPAID	For future use
TELEPHONY_PREPAID	For future use

12.4.73 servicePlanTypeEnum

Used for identifying the type of service plan. Values are of type String. Valid Values are:

Value	Description
INDIVIDUAL	Contracts created on individual plan will be standalone
SHARED	Service plan is part of a pool

12.4.74 serviceProviderProfileConfigTypeEnum,

Used for identifying OpenPort Simlock status values. Values are of type String. Valid Values are:

Value	Description
ATS	
BURST_DEVICE	
BURST_SERVICE	
CERTUS	
CERTUS_AV	
CERTUS_LM	
CERTUS_MAR	
CERTUS_VAM	
MACHINE_TO_MACHINE	
OPEN_PORT	
PAGING	
PTT_DEVICE	
PTT_TALKGROUP	
SCRATCHARD	
SHORT_BURST_DATA	
TELEPHONY	
TELPREPAID	

BETA

12.4.75 serviceTypeEnum

Used for specifying service type values. Values are of type String. Valid Values are:

Value	Description
ATS	This is used to locate ATS
BURST_DEVICE	This is used to locate BURST DEVICE accounts
BURST_SERVICE	This is used to locate BURST SERVICE accounts
CERTUS	This is used to locate CERTUS SERVICE
MACHINE_TO_MACHINE	This is used to locate MACHINE TO MACHINE (M2M)
OPEN_PORT	This is used to locate OPEN PORT (OP)
PAGING	This is used to locate PAGING
PTT_DEVICE	This is used to locate PUSH-TO-TALK DEVICE
PTT_TALKGROUP	This is used to locate PUSH-TO-TALK TALKGROUP
SCRATCHCARD	This is used to locate SCRATCH CARD (SC)
SHORT_BURST_DATA	This is used to locate SHORT BURST DATA (SBD)
TELEPHONY	This is used to locate TELEPHONY (TEL)
TELPREPAID	This is used to locate PREPAID

12.4.76 simLockStatusEnum

Used for identifying OpenPort Simlock status values. Values are of type String. Valid Values are:

Value	Description
LOCKED	Simlock is currently enabled.
UNLOCKED	Simlock is not currently enabled.

12.4.77 spnetGroupStatusEnum

Used for identifying status of SPNet Groups. Values are of type String. Valid Values are:

Value	Description
ACTIVE	Group is currently active and available for user assignment
OBSOLETE	Group is obsolete and no longer available for user assignment. Currently assigned users belonging to this group will retain permissions granted to the group.

12.4.78 statusChangeTypeEnum

Used for identifying value affected by type of status change. Values are of type String. Valid Values are:

Value	Description
CAPCODE	
IMEI	
ISDNA	
MSISDN	
SIM	
SUBSCRIBER_ACCOUNT_NUMBER	

BETA

12.4.79 statusEnum

Used for setting user status values. Values are of type String. Valid Values are:

Value	Description
ACTIVE	
CLOSED	
INACTIVE	

12.4.80 submarketTypeEnum

Used for submarket types. Values are of type String. Valid Values are:

Value	Description
ATS	
CREW	
POSTPAID	
PREPAID	
SAFETY	
GMDSS	

12.4.81 subscriberAccountSearchTypeEnum

Used for account searches. Values are of type String. Valid Values are:	Description
ACCOUNT_NUMBER	Search by account number
ATS_ICAO	Search by ICAO for ATS accounts
ATS_REG_TAIL_NUMBER	Search by registration/Tail Number for ATS accounts
BURST_SERVICE	Search for accounts subscribed to burst service by name.
CAPCODE	Search by Capcode value
DELIVERY_DEST	Search by SBD delivery destination settings
IMEI	Search by IMEI. For specific service IMEI values specify the serviceType value in the request.
IMO	Search by IMO for Certus Maritime accounts
IMSI	Search by IMSI associated to a sim
IP_ADDRESS	Search by IP address
ISDNA	Search by ISDNA for pager accounts
LOCAL	Search by local number value (PLUS 1)
MMSI	Search by MMSI value associated to maritime accounts (Certus only at this time)
MODEL	Search by device model (Certus only at this time)
MSISDN	Search by MSISDN value
MSISDNB	Search by MSISDNB value
MSISDNC	Search by MSISDNC value
MT_FILTER	Search by SBD MT Filter settings
PROVIDER_REFERENCE	Search by provider Reference value
PTT_TG	Search by talkgroup ID (issued by controller)
SHARED_GROUP	Search by Group account number
SIM_SERIAL_NUMBER	Search by sim serial (ICCID)
SMART	IWS will try to determine what to search on (based on the value passed in)
VERTICAL	Search by vertical market (Certus only at this time)
VESSEL_NAME	Search by Vessel name
VESSEL_TYPE	Search by Vessel type

12.4.82 transceiverTypeEnum

Used indicate if a model transceiver type. Values are of type string. Valid Values are:

Value	Description
BCX	
SFX	

12.4.83 telephonyScratchCardSearchTypeEnum

Used for specifying values for telephony scratchcard search types. Values are of type string. Valid Values are:

Value	Description
ACT_DATE	
BALANCE	
BATCH_NUMBER	
DISTR_TYPE	
EXPIRY_DATE	
FIRST_USE_DATE	
SERIAL_NUMBER	

BETA

12.4.84 userStatusEnum

Used for identifying user status values. Values are of type String.

Valid Values are:

Value	Description
ACTIVE	User account is active.
DEACTIVE	User account has been deactivated.
EXPIRED	User account is expired and requires password reset.
LOCKED	User account is locked due to invalid login attempts.

12.4.85 vesselLinkTypeEnum

Used for identifying the type of link between a vessel and a contract. Values are of type String. Valid Values are:

Value	Description
PRIMARY	
SECONDARY	
NON_SAFETY	

12.4.86 vesselTypeEnum

Used for identifying vessel types for Certus Maritime contracts. Values are of type String.

Valid Values are:

Value	Description
AHTS	
BulkCarrier	
Container	
Cruise	
Ferry	
FV	
LPG_LNG	
MSV	
Navy	
OSV	
RSV	
RORO	
Tanker	
Yacht	
Other	

12.4.87 voicePrepaidOptionsEnum

Used for specifying type of voice prepaid available to a model. Values are of type String. Valid Values are:

Value	Description
NONE	No prepaid permitted
SCRATCHCARD	Only scratchcard prepaid permitted
SIM	Only SIM based prepaid permitted
SCRATCHCARD_AND_SIM	Both scratchcard and SIM based prepaid permitted

12.4.88 voiceQualityEnum

Used for specifying the voice line quality available for a model. Values are of type String. Valid Values are:

Value	Description
STANDARD_ONLY	Standard voice quality only
HIGH_ONLY	High voice quality only
BOTH_MIXED	Allows both standard and high quality only
BOTH_SAME	Allows either standard or high quality but must be used on all voice lines.

12.4.89 voucherActionTypeEnum

Used for specifying values for prepaid voucher action types. Values are of type String.

Valid Values are:

Value	Description
ACTIVATE	This is used to indicate ACTIVATE request

BETA

RECHARGE	This is used to indicate RECHARGE request
REFUND	This is used to indicate REFUND request

12.4.90 voucherInventoryTypeEnum

Used for identifying voucher type values. Values are of type String. Valid Values are:

Value	Description
EVOUCHER	E-Voucher orders
EVOUCHER_RECHARGE	E-voucher recharge orders.
SCRATCHCARD	Scratch card orders

12.4.91 voucherOrderSearchTypeEnum

Used for specifying voucher order search type values. Values are of type String. Valid Values are:

Value	Description
BATCH_ID	Search by Batch ID
PURCHASE_ORDER	Search by PO number
SERIAL_NUMBER	Search by scratch card serial number

BETA

13 SOAP Methods

13.1 accountSearch

The account search can look up one or more accounts based on various search criteria. It supports wildcard (* or %) searches as well as status filters.

The accountSearch request consists of the following pieces of data. (accountSearchRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceTypeEnum)	No	Service Type filter for the request. If none is specified, it will be ignored. See Service Types
status (type:accountStatusEnum)	No	Optional filter used to narrow results down by account status. If none is specified, it will be ignored. See Account Status.
secondarySvcType (type: secondarySvcTypeEnum)	No	Secondary Service Type for the request. If none is specified, it will be ignored. See Secondary Service Types
secondarySvcStatus (type: secondarySvcStatusEnum)	No	Optional filter used to narrow results down by secondary service status. If none is specified, it will be ignored. See Secondary Service Status
dateSearchType (type: dateRangeTypeEnum)	No	Optional filter used to narrow results down by activation date range. If none is specified, it will be ignored. See Date Range Search Types
startDate (type:string)	Yes	This is the filter start date based on the subscriber activation date. This value will only be considered if dateSearchType = 'CUSTOM'
endDate (type:string)	Yes	This is the date till which the subscriber should be displayed. This value will only be considered if dateSearchType = 'CUSTOM'

BETA

Element/Field	Required?	Description
Seconda00rySvcStatus (type: secondarySvcStatusEnum)	No	Optional filter used to narrow results down by secondary service status. See Secondary Service Status
filterType (type: subscriberAccountSearchTypeEnum)	No	Which method should be used to lookup the account. See Subscriber Account Search Type . NOTE: is filter selection is desired user must supply a value in each of filterType, filterCond and filterValue
filterCond (type: searchCondEnum)	No	Which type of match should be used for search filter. See Filter Conditions. NOTE: is filter selection is desired user must supply a value in each of filterType, filterCond and filterValue
filterValue (type:string)	No	Based on the filter type specified, this should be the value representing the account to search for. NOTE: is filter selection is desired user must supply a value in each of filterType, filterCond and filterValue
limit (type:int)	Yes	Number of records to be retrieved in a single call. The default limit is set to 5000 records.
startRow (type:int)	Yes	Start record that the query will be based off. This is used in combination with the “limit” field.

BETA

A successful search will return a list of results that the service provider specified has access to. The list will be empty if there are no devices matching the criteria.
The response will contain the following: (accountSearchResponseImpl)

Element/Field	Required?	Description
totalNumberOfRecords (type:int)	No	This is the total number of results that matched your search criteria. This value may be larger than the actual results

Element/Field	Required?	Description
		returned based on limit parameter supplied on request.
Subscribers	No	Array of subscriber details matching search criteria
subscriber (type: accountSearchReportImpl)	No	This object contains the subscriber account information
accountNumber (type:string)	Yes	Contains the actual account number which represents the subscriber account. Follows the convention “SUB-XXXXXXX”.
serviceType (type:string)	No	See Service Types.
details (type:serviceTypeEnum)	Yes	Identifier for the account requested. The type of information will vary based on filter type requested. Default values: OpenPort: SIM (ICCID) Telephony: SIM (ICCID) Certus: SIM(ICCID) ATS: SIM (ICCID) SBD:IMEI M2M:IMEI PTT Device: IMEI PTT Talkgroup:Talkgroup ID Burst Service:Service Name Burst Device:IMEI Paging:Capcode If a specific filter type is used, the value corresponding to that type will be returned in details. For example search by vessel name will return vessel name values.
activationDate (type:string)	No	Date of activation
lastUpdated (type:string)	Yes	This is the date the account was last modified by user request. It is a system generated field. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
accountStatus (type:accountStatusEnum)	Yes	Contains the current status of the subscriber account. See Account Status.

BETA

Element/Field	Required?	Description
planName (type: string)	Yes	The name of the plan the account has selected
iccid (type: string)	No	If applicable the iccid will be included
Imei (type: string)	No	If applicable the IMEI will be included
spReference (type: string0	No	If available, the provider reference value will be included.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=accountSearch

BETA

Errors or exceptions

Error	What it means/How to fix it
Either did not find a unique account matching search type [<search type>] and value [<value to search for>] or SP [XXXXXX] does not have access.	This error is returned for one of 3 reasons: - The account doesn't exist. - The search criteria don't match up (i.e., the search type is SUBSCRIBER_ACCOUNT_NUMBER, but a MSISDN was passed). - The SP specified by the serviceProviderAccountNumber does not have access to the subscriber being queried.
Wildcards are not supported for SMART searches.	If SMART is specified as the search type, then wildcards cannot be used.
Unable to determine search type for value [<value to search for>].	If SMART is specified as the search type, then IWS will attempt to figure out what the user is trying to search by based on the value passed in. If it is unable to do so (i.e., doesn't match any of the expected regular expressions), then this error will be returned.
Unable to find parent account [XXXXXX]. No results to return...	The service provider specified by the serviceProviderAccountNumber on the request could not be located. Thus, access to search results cannot be verified and none will be returned. Please check to ensure that you have specified the correct value.
Unsupported search type [<search type>]	As functionality is still being added to the web services to support all of Iridium's offerings, some of the search types are not yet supported. If you get this error back, please ensure that you are passing in one of the supported options specified in the request details above.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.2 accountUpdate

The accountUpdate request consists of the following pieces of data. (accountUpdateRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Account Object	Yes	Specify one type of Account object to be considered for this request. See Account for details regarding the details for each account type.
openPortSubscriberAccount2 (type: openPortSubscriberAccountImpl)	No	Use if modifying an OpenPort account
telephonySubscriberAccount2 (type: telephonySubscriberAccountImpl)	No	Use if modifying an Telephony account
pagingSubscriberAccount2 (type: pagingSubscriberAccountImpl)	No	Use if modifying an Paging account
atsSubscriberAccount2 (type: atsSubscriberAccountImpl)	No	Use if modifying an ATS account
burstDeviceSubscriberAccount2 (type: burstDeviceSubscriberAccountImpl)	No	Use if modifying an Burst Device account
sbdSubscriberAccount2 (type: sbdSubscriberAccountImpl)	No	Use if modifying an SBD account
m2mSubscriberAccount2 (type: m2MSubscriberAccountImpl)	No	Use if modifying an M2M account
pttDeviceSubscriberAccount2 (type: pttDeviceSubscriberAccountImpl)	No	Use if modifying an PTT Device account
pttTalkGroupSubscriberAccount2 (type: pttTalkGroupSubscriberAccountImpl)	No	Use if modifying an PTT Talkgroup account
certusSubscriberAccount2 (type: certusSubscriberAccountImpl)	No	Use if modifying an Certus account

If the request is successful, a response will be returned with the fully-updated account on it. (accountUpdateResponseImpl)

Element/Field	Required?	Description
Account Object	Yes	One of the following account objects will be returned for a successful request.

BETA

Element/Field	Required?	Description
		See Account for details regarding the details for each account type.
openPortSubscriberAccount (type: openPortSubscriberAccountImpl)	No	Provided if account type was Open Port
telephonySubscriberAccount (type: telephonySubscriberAccountImpl)	No	Provided if account type was Telephony
pagingSubscriberAccount (type: pagingSubscriberAccountImpl)	No	Provided if account type was Paging
atsSubscriberAccount (type: atsSubscriberAccountImpl)	No	Provided if account type was ATS
burstDeviceSubscriberAccount (type: burstDeviceSubscriberAccountImpl)	No	Provided if account type was Burst Device
sbdSubscriberAccount (type: sbdSubscriberAccountImpl)	No	Provided if account type was SBD
m2mSubscriberAccount (type: m2MSubscriberAccountImpl)	No	Provided if account type was M2M
pttDeviceSubscriberAccount (type: pttDeviceSubscriberAccountImpl)	No	Provided if account type was PTT Device
pttTalkGroupSubscriberAccount (type: pttTalkGroupSubscriberAccountImpl)	No	Provided if account type was PTT Talk Group
certusSubscriberAccount (type: certusSubscriberAccountImpl)	No	Provided if account type was Certus

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=accountUpdate

Errors or exceptions

Error	What it means/How to fix it
NOTE: Many of the same error conditions listed above for activations apply to updates as well. The following list will note error conditions specific to updates. Please see Activation errors for any errors not listed here.	
The Account you are trying to update does not exist.	The account identified by the subscriberAccountNumber passed in on the Account object could not be located in the system. Please verify that the account number is correct and that it is one of the accounts that the service provider has access to.
Service provider [<serviceProviderAccountNumber>] is not a parent of account [<subscriberAccountNumber >].	The specified account is not owned by the current service provider. Please insure that both account numbers are correct.
No updates are allowed while the account status is <PENDING/SUSPENDED/DEACTIVE>.	Subscriber accounts cannot be updated while they are in a PENDING, SUSPENDED, or DEACTIVE state. Please update the status to ACTIVE prior to making modifications to the account. (See How to change the status of an OpenPort account.)
The new status was null. Please specify a valid status.	The Account status value was not set. Please specify the desired ending status.
Cannot set account status to PENDING. This is a derived state.	The Account status value was specified as PENDING. The status cannot be directly set to PENDING. Please specify either ACTIVE, DEACTIVE, SUSPENDED, or CLOSED..
No status changes allowed on a CLOSED account.	The specified account is currently CLOSED. No further updates are allowed to closed accounts.
No status changes are allowed while service orders are still pending.	A previous update to this account resulted in the generation of one or more provisioning service orders. These service orders have not yet completed. No additional status changes are allowed until the pending service orders complete. Please retry the request later.

BETA

Error	What it means/How to fix it
There are transitions scheduled for the account. You must cancel these prior to updating other price-related features.	Previously, a change involving price-related features has been made to the account. That change is scheduled to take effect in the future. Other price-related changes are prohibited until the pending changes either take effect or are cancelled (see How to Cancel Scheduled Transitions).
Unknown Error. OR Cannot remove members from incremental discount group. SP is null. OR Cannot remove members from incremental discount group. Group is CLOSED. OR Failed to remove account from incremental discount group. Account Number: <accountNumber> not in group.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.3 accountUpdateDevices

The accountUpdateDevices is used for swapping device resources on an account. This method can be used to swap SIM, MSISDN, MSISDN-C, +1/+7 numbers, Capcodes or ISDNA values. Multiple devices on an account may be provided in a single call. New devices/numbers must belong to the same submarket as the original devices/numbers on the account. If a change to submarket is desired, the account must be deactivated and reactivated using the new device and submarket. This method is currently only supported for Telephony and Paging accounts. The request consists of the following pieces of data. (accountUpdateDevicesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
subscriberAccountNumber (type: string)	Yes	Specify account number to be modified
Devices	Yes	Array of devices to be changed (only supply those devices that are to be modified)
device (type: deviceSwapImpl)	Yes	
fromDevice (type: deviceImpl)	Yes	Specify old device number, device ID can be left blank

BETA

Element/Field	Required?	Description
toDevice (type:deviceImpl)	No	Specify new device number, device ID can be left blank Required for SIM swap. To have numbers autogenerated leave empty (MSISDN, MSISDN-C, ISDN-A) For +1/+7 numbers leave blank, any value provided will be ignored as these numbers must be autogenerated. For IP swap send toVpnDevice and leave toDevice blank For Local Number (PLUS1) swap send toLocalCountry and leave toDevice blank
toVpnDevice (type: deviceIpImpl)	No	Specify new device information for IP swap if VPN change is desired.
toLocalCountry (type: localCountryImpl)	No	Specify new device information for local number swap if change in country is desired.
homeGateway (type: homeGatewayEnum)	No	Optional field to specify the home gateway for SIM-based IMEIs if the customer has privileges to override the default home gateway specified in the service provider profile.

BETA

If the request is successful, an empty response will be returned.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=accountUpdateDevices

Errors or exceptions

Error	What it means/How to fix it
NOTE: Many of the same error conditions listed above for activations apply to updates as well. The following list will note error conditions specific to updates. Please see Activation errors for any errors not listed here.	

Error	What it means/How to fix it
The Account you are trying to update does not exist.	The account identified by the subscriberAccountNumber passed in on the Account object could not be located in the system. Please verify that the account number is correct and that it is one of the accounts that the service provider has access to.
Service provider [<serviceProviderAccountNumber>] is not a parent of account [<subscriberAccountNumber >].	The specified account is not owned by the current service provider. Please insure that both account numbers are correct.

13.4 accountUpdateHomeGateway

The accountUpdateHomeGateway call is used to change the provisioned location for a device from one home gateway to another. This method is not used for swapping devices. This method is currently only supported for SBD and M2M accounts. The request consists of the following pieces of data. (accountUpdateHomeGatewayRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
accountNo (type: string)	Yes	Specify account number to be modified
homeGateway (type: homeGatewayEnum)	Yes	Field to specify the new home gateway for SIM-based IMEIs. Customer must have privileges to override the default home gateway specified in the service provider profile.

If the request is successful, an empty response will be returned.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=accountUpdateHomeGateway

13.5 activateScratchCards

The activateScratchCards request consists of the following pieces of data.   This call is used to activate OpenPort scratch cards. (activateScratchCardRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
type (type:activationTypeEnum)	Yes	See Activation Types.

BETA

Element/Field	Required?	Description
lineItemId (type:string)	Yes	Line item identifier. Used when doing a LINEITEM activation.
startSerial (type:string)	No	Starting serial number. Used when doing a RANGE activation.
endSerial (type:string)	No	Ending serial number. Used when doing a RANGE activation.
validateOnly (type:boolean)	Yes	Flag used to validate an activation request without actually causing it to process.
purchaseOrder (type:string)	No	Purchase order identifier. Used for FULL activations.

If the request is successful, a response will be returned with the following data. (scratchCardActivationResponseImpl)

Element/Field	Required?	Description
purchaseOrder (type:string)	Yes	Purchase order identifier associated with the activation request.
requestId (type:string)	Yes	A number to identify the activation request.
serialNumbers	No	An array to hold the serial numbers of scratch cards activated with this request.
item (type:string)	No	A serial number.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=activateScratchCards

Errors or exceptions

Error	What it means/How to fix it
No results returned. Scratchcards have already been activated.	All scratch cards requested have already been activated.
Error activating scratch card batch.	A system error was returned by the billing system. Try your activation again. If the error persists, contact Iridium Account Services to investigate.
	If this error is returned, there was some sort of a system failure or exception.

BETA

Error	What it means/How to fix it
Unknown Error	Please contact Iridium Service Delivery so they can investigate.

13.6 activateTelephonyScratchCards

The activate telephony scratch cards allows a service provider to activate a single card or a range of scratch cards by the service provider, startSerialNumber, endSerialNumber, and batchNumber, or any combination of those.

The activateTelephonyScratchCards request consists of the following pieces of data. (activateTelephonyScratchCardsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
startSerial (type:string)	No	Starting serial number. Used when doing a RANGE activation.
endSerial (type:string)	No	Ending serial number. Used when doing a RANGE activation.
batchNumber (type: string)	No	Value of scratchcards that is assigned/sold to an SP.

If the request is successful, a response will be returned with the following data. (activateTelephonyScratchCardsResponseImpl)

Element/Field	Required?	Description
success (type: boolean)	Yes	Result of activated telephony scratch cards account request. True – Success False – Failed
message (type: string)	Yes	
startSerial (type: string)	Yes	Starting serial number. Used when doing a RANGE activation.
endSerial (type: string)	Yes	Ending serial number. Used when doing a RANGE activation.
batchNumber (type: string)	Yes	Value of scratchcards that is assigned/sold to an SP.

BETA

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=activateTelephonyScratchCards

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error activating scratch card batch.	If this error is returned, there was a problem with the Activate Telephony Scratch Cards IWS call. Please contact Iridium Service Delivery so they can investigate..
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.7 activateSubscriber

The activateSubscriber request consists of the following pieces of data. (activateSubscriberRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Account Object	Yes	Specify one type of Account object to be considered for this request. See Account for details regarding the details for each account type.
openPortSubscriberAccount (type: openPortSubscriberAccountImpl)	No	Provided if account type was Open Port
telephonySubscriberAccount (type: telephonySubscriberAccountImpl)	No	Provided if account type was Telephony
pagingSubscriberAccount (type: pagingSubscriberAccountImpl)	No	Provided if account type was Paging
atsSubscriberAccount (type: atsSubscriberAccountImpl)	No	Provided if account type was ATS
burstDeviceSubscriberAccount (type: burstDeviceSubscriberAccountImpl)	No	Provided if account type was Burst Device

Element/Field	Required?	Description
sbdSubscriberAccount (type: sbdSubscriberAccountImpl)	No	Provided if account type was SBD
m2mSubscriberAccount (type: m2MSubscriberAccountImpl)	No	Provided if account type was M2M
pttDeviceSubscriberAccount (type: pttDeviceSubscriberAccountImpl)	No	Provided if account type was PTT Device
pttTalkGroupSubscriberAccount (type: pttTalkGroupSubscriberAccountImpl)	No	Provided if account type was PTT Talk Group
certusSubscriberAccount (type: certusSubscriberAccountImpl)	No	Provided if account type was Certus

If the request is successful, a response will be returned with the fully-populated account including the account number that was assigned to the subscriber, the generated fields such as the created date and current status of the account.

Element/Field	Required?	Description
Account Object	Yes	The activated account object will be returned. See Account for details regarding the details for each account type.
openPortSubscriberAccount (type: openPortSubscriberAccountImpl)	No	Provided if account type was Open Port
telephonySubscriberAccount (type: telephonySubscriberAccountImpl)	No	Provided if account type was Telephony
pagingSubscriberAccount (type: pagingSubscriberAccountImpl)	No	Provided if account type was Paging
atsSubscriberAccount (type: atsSubscriberAccountImpl)	No	Provided if account type was ATS
burstDeviceSubscriberAccount (type: burstDeviceSubscriberAccountImpl)	No	Provided if account type was Burst Device
sbdSubscriberAccount (type: sbdSubscriberAccountImpl)	No	Provided if account type was SBD
m2mSubscriberAccount (type: m2MSubscriberAccountImpl)	No	Provided if account type was M2M

DRAFT

Element/Field	Required?	Description
pttDeviceSubscriberAccount (type: pttDeviceSubscriberAccountImpl)	No	Provided if account type was PTT Device
pttTalkGroupSubscriberAccount (type: pttTalkGroupSubscriberAccountImpl)	No	Provided if account type was PTT Talk Group
certusSubscriberAccount (type: certusSubscriberAccountImpl)	No	Provided if account type was Certus

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=activateSubscriber

Errors or exceptions

Error	What it means/How to fix it
The account to activate was null	The openPortSubscriberAccount passed in on the request was null/empty. This is a required field and should contain all of your desired Account data.
The plan for activation was null	The plan object on the openPortSubscriberAccount passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Service provider has not been configured to allow non-production accounts.	Somehow, the request attempted to activate a subscriber as a DEMO account; however, the service provider does not have any configured. If this is in error, please contact Iridium Service Delivery.
DEMO Account Limit Exceeded for Service Provider: XXXXXX.	The service provider associated with account number XXXXXX does not have any more DEMO accounts to offer. Before activating any new subs under these plans, you must either deactivate an existing one.
The SIM device is null or empty. This field is required for subscriber activation.	The SIM is a required field. Please populate it on the Account.

BETA

Error	What it means/How to fix it
The SIM device was not found [<deviceNumber>].	The SIM specified on the Account could not be located by the device number specified. Please ensure that the SIM you are trying to use is valid. If you believe it is valid (and in a RESERVED state), please contact Iridium Service Delivery so they can investigate.
At least one line is required for account activation.	There must be at least one phone line (Number) on an Account. Please ensure one is specified.
The MSISDN specified was not found {<deviceNumber>].	The Number specified as one of the phone lines on the Account could not be located by the device number specified. Please ensure that the number you are trying to use is valid. If you believe it is valid (and in a RESERVED state), please contact Iridium Service Delivery so they can investigate.
Device number is null.	This error can be thrown in any case where the device number specified on a device object is null. Verify that all required deviceNumber fields are populated.
Invalid IMSI [<imsi>].	The IMSI specified is not a 15 digit numerical value.
Invalid SIM (ICCID) [<sim>].	The SIM device number specified is not a 19 digit numerical value.
Invalid MSISDN [<msisdn>].	The MSISDN device number (phone line) specified is not a 12 digit numerical value starting with 8816 or 8817.
Invalid IP address [<ip address>].	The IP address device number specified does not follow an acceptable pattern for IP addresses (i.e., 255.255.255.255 or 127.0.0.1).
Device [<device number>] does not belong to account = [<requesting service provider>].	This error is thrown when a device is being activated (or swapped) on an account, but the device is not part of the service provider's pool of associated devices. This error would indicate that the device number is incorrect or the devices have not been properly associated in the system. In the latter case, contact Iridium Service Delivery for further investigation.

BETA

Error	What it means/How to fix it
Invalid sub market for SIM device [<device number>] subMarket = [<sub market>] serviceType = [<service type>].	For OpenPort, the valid sub markets are 703 and 777. This number appears in the SIM number itself (as well as the corresponding Numbers on the account). Otherwise, this error will be thrown.
SIM network [<market>] is different than the current network [<current market>].	When swapping the SIM on an account, the new market must match the existing market of the account. Otherwise, this error is thrown.
Network [<network>] does not match SIM network [<sim network>].	The network of a Number must match the network of the SIM that it is being activated on. Otherwise, this error is thrown.
Invalid state for device [<device number>] state = [<state>].	This error is thrown when a device is being activated (or swapped) on an account. For this to succeed, the device must be in a RESERVED state. Otherwise, this error will be thrown.
Invalid voice channel [<voice channel>].	Please refer to Number for the business rules around voice channel.
Invalid line type [<line type>].	Please refer to Number for the business rules around line type.
Invalid two stage PIN [<two stage pin>].	Please refer to Number for the business rules around two stage.
Invalid plus one number [<plus one number>] or plus one is not allowed for line type [<line type>].	Please refer to Number for the business rules around plus one.
Invalid captain PIN [<captain pin>] or captain PIN is not allowed for line type [<line type>].	Please refer to Number for the business rules around captain PIN.
Voicemail is not allowed on line type [<line type>].	Please refer to Number for the business rules around voicemail.
Too many programmable buttons were specified or they had the wrong value [<programmable buttons>].	Please refer to Number for the business rules around programmable buttons.
The bundle identified by ID [<bundleId>] does not exist.	A bundle specified on the Plan is unknown or not applicable. This probably means that the Utility Services were not properly utilized to make the plan bundle selections initially.

BETA

Error	What it means/How to fix it
Bundle [<details about bundle being validated>] is invalid for the plan selections.	A bundle specified on the Plan is invalid for that Plan. (The error detail will provide an indication of which bundle is in error.) This probably means that the Utility Services were not properly utilized to make the plan bundle selections initially.
One or more errors were encountered during activation post processing: Error count [X]. <List of specific errors encountered>	This message will be returned if any issues are encountered while performing the following actions: - Updating the DEMO counts. When this occurs, it means that the subscriber was activated successfully; however, the follow up activities may not have completed. Contact Iridium Service Delivery to investigate.
Unknown Error. OR Unable to locate activation fee for SP: <serviceProviderAccountNumber> OR There is more than 1 activation fee configured for SP: <serviceProviderAccountNumber>. OR An error occurred while adding account to incremental discount group.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.
Error Gathering Demo/Trial Bundles	If this error is returned, there was a problem with the demo/trial IWS call. Please contact Iridium Service Delivery so they can investigate
Error Gathering Promo Bundles	If this error is returned, there was a problem with the Promo IWS call. Please contact Iridium Service Delivery so they can investigate
Error Gathering Service Plan Bundles	If this error is returned, there was a problem with the Service Plan IWS call. Please contact Iridium Service Delivery so they can investigate
Error Gathering Language Bundles	If this error is returned, there was a problem with the Language IWS call. Please

BETA

Error	What it means/How to fix it
	contact Iridium Service Delivery so they can investigate

13.8 addDIDWWNumberInventory

The addDIDWWNumberInventory request is used to purchase DIDWW numbers from outside vender to be used as local numbers. The request consists of the following pieces of data. (addDIDWWNumberInventoryRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See DIDWW Authentication.
countryId (type:boolean)	Yes	See getLocalNumberRegions
count (type: int)	Yes	Quantity for the order

If the request is successful IWS will return a response object with a list of generated local numbers. (addDIDWWNumberInventoryResponseImpl)

Element/Field	Required?	Description
orderId (type: string)	Yes	Unique id for requested order from DIDWW vendor
successfulLocalNumbers	No	An array of devices that successfully populated in the billing system.
localNumber (type: string)	No	
failedLocalNumbers	No	An array of devices that failed to populate in the billing system.
localNumber (type: string)	No	
errorMessage (type:string)	No	Provides error message should any devices fail to populate in billing system.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=addDIDWWNumberInventory

13.9 associateSimToImeiDevice

The associateSimToImeiDevice request is used for M2M device manufacturers to enable linking of SIM to IMEI once soldered together onto a single unit. The request consists of the following pieces of data. (associateSimToImeiRequestImpl)

DRAFT

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
bulkAssociate (type:boolean)	No	Indicates a bulk request
devicePairs	No	An array of device pairings
devicePair (type: devicePairImpl)	Yes	
deviceSim (type:deviceSimImpl)	Yes	See DeviceSIM. This is the object for sim being linked to the IMEI.
deviceImei (type:deviceImeiImpl)	Yes	See DeviceIMEI. This is the object for IMEI being linked to the SIM.

If the request is successful IWS will return an empty response object. If this is a bulk request the response will return a bulk ID. (associateSimToImeiResponseImpl)

Element/Field	Required?	Description
Bulk Request Id (type: string)	No	The request ID for the bulk transaction will be returned.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=associateSimToImeiDevice

BETA

13.10 authenticatedPing

The authenticatedPing request is used to validate connection to IWS.

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.

If the request is successful, an empty response will be returned.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=authenticatedPing

13.11 bulkAccountUpdate

The bulkAccountUpdaterequest consists of the following pieces of data. (bulkAccountUpdateRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Bulk Plan Update Object	Yes	Specify one type of bulk plan update objects to be considered for this request. sbdBulkPlanUpdate: sbdBulkPlanUpdateImpl m2mBulkPlanUpdate: m2MBulkPlanUpdateImpl telephonyBulkPlanUpdate: telephonyBulkPlanUpdateImpl pttDeviceBulkPlanUpdate: pttDeviceBulkPlanUpdateImpl See Bulk Plan Update for details regarding the details for each type.

If the request is successful, a response will be returned with the bulk request ID. This ID can be utilized to request a report detailing the status for each request in the bulk object. (bulkAccountUpdateResponseImpl)

Element/Field	Required?	Description
Bulk Request Id (type: string)	No	The request ID for the bulk transaction will be returned.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=bulkAccountUpdate

BETA

13.12 bulkActionSearch

The bulkActionSearch Request consists of the following pieces of data. (bulkActionSearchRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
bulkRequestId (type: string)	Yes	Search string to identify request id
Wildcard (type:Boolean)	No	TRUE – treat request ID as a wildcard search parameter. FALSE – search for specific request ID provided
serviceType (type: serviceTypeEnum)	No	Service Type for the request. See Service Types
isDeviceTransfer (type: Boolean)	No	TRUE – process request as a bulk device transfer request FALSE – do not process as bulk device transfer
requestType (type: string)	No	Filter results by type of bulk request.

If the request is successful, a response will be returned with the listing of bulk requests. The ids can be utilized to request a report detailing the status for each request in the bulk object. (bulkActionSearchResultsImpl)

Element/Field	Required?	Description
results (type: bulkActionSearchResultsImpl)	Yes	An array of bulk actions that meet the search criteria provided
Bulk Actions (type: bulkActionSearchResultsImpl)	No	An array of bulk actions that meet the search criteria provided
bulkAction (type: bulkActionEntryImpl)	No	
bulkActionId (type:string)	No	Bulk request ID
serviceType (type: serviceTypeEnum)	No	Service Type for the request. See Service Types
status (type:string)	No	Status of the request
createdDate (type:string)	No	Date request was submitted
modifiedDate (type:string)	No	Last date of request modification
requestType (type:string)	No	Type of bulk request

BETA

Element/Field	Required?	Description
configuredResultLimit (type: int)	No	Result set limit
totalResultCount (type: int)	No	Total results returned

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=bulkActionSearch

13.13 bulkAddPrepaid

The bulkAddPrepaid request allows the addition of crew or prepaid network object to an existing contract.

The bulkAddPrepaid request consists of the following pieces of data: (bulkAddPrepaidRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
planType (type: string)	Yes	See bulkPrepaidTypeEnum
telephonyImsis	Yes	An array of device imsis
imsi (type:string)	Yes	
planId (type: userStatusEnum)	No	Prepaid plan value.
prepaidVoucher (type: BulkPrepaidVoucher)	No	See Prepaid Voucher
spAlert (type: boolean)	No	Alert service provider for balance expiration. Expiration and Balance alert warnings TRUE – Enabled FALSE – Disabled Value Defaults to FALSE
restrictScratchCards (type: boolean)	No	True/False Is restrict flag set
language (type: prepaidLanguageTypeEnum)	Yes	Announcement language. See Prepaid Language Type Enum.

A typical response will contain the following. (bulkAccountUpdateResponseImpl)

BETA

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=bulkAddPrepaid

BETA

13.14 bulkRechargePrepaidAccount

The bulkAddPrepaidAccount request allows voucher recharges prepaid accounts that already exist with Prepaid network access.

The bulkRechargePrepaidAccount request consists of the following pieces of data: (bulkRechargePrepaidAccountRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
telephonyImsis	Yes	An array of device imsis
imsi (type:string)	Yes	
planId (type: userStatusEnum)	Yes	
prepaidVoucher (type: BulkPrepaidVoucher)	Yes	See Prepaid Voucher
spAlert (type: boolean)	No	Alert service provider for balance expiration. Expiration and Balance alert warnings TRUE – Enabled FALSE – Disabled Value Defaults to FALSE
language (type: prepaidLanguageTypeEnum)	Yes	Announcement language. See Prepaid Language Type Enum.

A typical response will contain the following. (bulkAccountUpdateResponseImpl)

To generate sample requests and responses: https://training.iridium.com/iwsexplorer/index.php?call=bulkRechargePrepaidAccount

13.15 changeUserProfileStatus

The changeUserProfileStatus request is used to update the status of an SPNet Pro user account.

The changeUserProfileStatus request consists of the following pieces of data: (changeUserProfileStatusRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
userId (type: string)	Yes	Id for user object to be updated.

BETA

Element/Field	Required?	Description
username (type: string)	Yes	The username (typically email address) for the account to be modified.
status (type: userStatusEnum)	Yes	The new status to set for the user account. See User Status.

If the request is successful, the account status will be modified and an empty response will be returned.

13.16 createSecondaryDataService

The createSecondaryDataService request is used to create new secondary data service objects used for Certus provisioning.

The createSecondaryDataService request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
secondaryDataService (type: secondaryDataServiceImpl)	Yes	The new Secondary Data Service to be created. See Secondary Data Service

A typical response will contain the following. (GetSecondaryDataServiceResponseImpl)

Element/Field	Required?	Description
secondaryDataService (type: secondaryDataServiceImpl)	No	See Secondary Data Service
permittedProviderAccounts	No	This is the list of providers that have access to the Secondary Data Service.
permittedProviderAccount (type: string)	No	Provider account number

13.17 createSubscriberOrganization

The Create Subscriber Organization method will enable to the user to create a new subscriber organization for use with the PTT Controller.

The createSubscriberOrganization request consists of the following pieces of data. (createSubscriberOrgRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.

BETA

subscriberOrganization (type: subscriberOrganizationImpl)	Yes	The subscriber organization details See Subscriber Organization

A request will return a fully-populated subscriber organization object as requested with an ID assigned. The response will contain the following: (subscriberOrgsResponseImpl)

Element/Field	Required?	Description
Subscriber Organizations	No	An array of subscriber organizations
subscriberOrganization (type:subscriberOrganizationImpl)	No	The subscriber organization details See Subscriber Organization
subscriberOrganizations		
subscriberOrganization (type: subscriberOrganizationImpl)	No	Array of subscriber organizations

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=createSubscriberOrganization

13.18 createUserProfile

The createUserProfile request is used to create a new user account for SPNetPro access.

The createUserProfile request consists of the following pieces of data: (createUserProfileRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
user (type:userImpl)	Yes	User Object. See User.

A typical response will contain the following (createUserProfileRequestImpl)

Element/Field	Required?	Description
user (type: userImpl)	Yes	The user object created. See User.

13.19 createVessel

The create vessel call is used to setup a new vessel object used for maritime provisioning.

BETA

The createVessel request consists of the following pieces of data: (createVesselRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
vessel (type: vesselImpl)	Yes	See Vessel

A typical response will contain the following. (createVesselResponseImpl)

Element/Field	Required?	Description
vessel (type: vesselImpl)	Yes	See Vessel

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=createVessel

13.20 deactivateTelephonyScratchcard

The deactivate telephony scratchcard allows a service provider to deactivate individual scratch cards the service provider, and serial number.

Note: Any balance will be lost as part of this request.

The deactivateTelephonyScratchCard request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serialNumber (type: string)	Yes	Serial number for telephony scratchcard to be deactivated.

A typical response will contain the following. (deactivateTelephonyScratchcardresponseimpl)

Element/Field	Required?	Description
deactivateTelephonyScratchCardImpl	Yes	
success (type: boolean)	Yes	Result of deactivated telephony scratchcard account request. True – Success False – Failed
message (type: string)	Yes	
serialNumber (type: string)	Yes	Serial number for deactivated telephony scratchcard.

BETA

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=deactivateTelephonyScratchCard

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.21 deletePrepaidAccount

The delete prepaid account allows a service provider to remove a prepaid account from an active subscriber.

Note: Any balance will be lost as part of this request.

The deletePrepaidAccount request consists of the following pieces of data: (deletePrepaidAccountRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
msisdn (type: string)	Yes	The msisdn of the prepaid account to be deleted.
iccid (type: string)	Yes	The iccid of the sim for the prepaid account to be deleted.

A typical response will contain the following. (deletePrepaidAccountResponseImpl)

Element/Field	Required?	Description
deletePrepaidAccountImpl	Yes	
msisdn (type: string)	Yes	The msisdn of the prepaid account to be deleted.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=deletePrepaidAccount

BETA

13.22 deleteSecondaryDataService

The deleteSecondaryDataService request is used delete an existing secondary data service. Note: This function will only be successful if the secondary data service is only associated to one SP account

The deleteSecondaryDataService request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
secondaryDataServiceId (type: string)	Yes	The internal id for secondary data service.

If the request is successful, an empty response will be returned.

13.23 deviceTransferOwner

The deviceTransferOwner method allows a customer to transfer ownership of a resource currently owned by the requesting Service Provider to another Service Provider. The resource must be currently available and not linked to an active or suspended contract. The resource must also be one managed in inventory (i.e. SBD IMEIs are managed inventory, where Telephony IMEIs are not). A user may not transfer ownership of a GLOBAL resource.

The deviceTransferOwner request consists of the following pieces of data: (deviceTransferOwnerRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
deviceStringType (type:deviceStringTypeEnum)	Yes	The type of device that will be checked against the string value of deviceString. See deviceStringTypeEnum.
deviceString (type:string)	Yes	The value of the device string (IMEI, SIM, etc.)
toSPAccountNumber	No	If specified this will be the account number for the new owner. If not specified the resource will be returned to the GLOBAL pool.
bulkAction (type: Boolean)	Yes	TRUE – indicates this request is for a bulk action FALSE – indicates this request is a single transaction

BETA

Element/Field	Required?	Description
bulkDetails	No	An array of device strings to be transferred
bulkDeviceString	Yes	The value of the device string.

A typical response will contain the following. (deviceTransferOwnerResponseImpl)

Element/Field	Required?	Description
deviceTransferResponse (type: string)	Yes	Result of deactivated telephony scratchcard account request. True – Success False – Failed

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=deviceTransferOwner

13.24 disAssociateSimFromImeiDevice

The disAssociateSimFromImeiDevice request is used for M2M device manufacturers to remove linking of SIM to IMEI. The request consists of the following pieces of data. (disAssociateSimFromImeiRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
deviceSim (type:deviceSimImpl)	Yes	See DeviceSIM. This is the object for sim being linked to the IMEI
deviceImei (type:deviceImeiImpl)	Yes	

If the request is successful, an empty response will be returned.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=disAssociateSimFromImeiDevice

13.25 findServiceProviderProfile

The findServiceProviderProfile request is used to retrieve the profile parameters for a service provider account.

The findServiceProviderProfile request consists of the following pieces of data: (searchSPProfileRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.

BETA

A typical response will contain the following (searchSPProfileResponseImpl)

Element/Field	**Required?**	**Description**
sp (type: serviceProviderImpl)	No	
accountNumber (type: string)	Yes	Service Provider Account number
accountName (type: string)	Yes	Service Provider Account Name
status (type:accountStatusEnum)	Yes	Account Status. See Account Status.
accountMgrId (type: string)	Yes	Account manager id.
plmnId (type: string)	Yes	Identifier for account files.
extractionFormat (type: extractionFormatEnum)	Yes	Type of daily CDR extract format to be provided.
trackingEnabled (type: boolean)	Yes	TRUE – tracking features are enabled for this SP. FALSE – tracking features are not available for this account.
reportingEnabled (type: boolean)	Yes	TRUE – scheduled reporting features are enabled for this SP. FALSE – scheduled reporting features are not available for this account.
exemptEnabled (type: boolean)	Yes	TRUE – FALSE –
lateFeePercent (type: string)	No	
paymentTerms (type: string)	No	
gracePeriod (type: string)	No	
deviceUsageType (type: deviceUsageTypeEnum)	No	See Device Usage Types
messageText (type: string)	No	
startDate (type: string)	No	
endDate (type: string)	No	
configList	No	Array of service configurations
config (type: spServiceConfigurationInfoImpl)	No	

BETA

Element/Field	Required?	Description
serviceType (type: serviceProviderProfileConfigTypeEnum)	Yes	See Service Provider Profile Types
supported (type: Boolean)	Yes	TRUE – service is enabled for this SP. FALSE – service is not available for this account.
demoOnly (type: Boolean)	Yes	TRUE – This is a demo only account, no commercial accounts for the service specified are allowed. FALSE – Commercial and/or demo account types are permitted.
allowAutoRecharge (type: Boolean)	Yes	TRUE – auto recharge is permitted FALSE – auto recharge is not permitted for this account and service.
defaultHomeGateway (type:homeGatewayEnum)	No	This value will specify the default home gateway to be used for all provisioning actions within the service for SIM-based IMEIs. NOTE: non SIM-based IMEIs have a hard-coded home gateway which cannot be changed
gatewayOverrideFlag (type:Boolean)	Yes	True: customer has the privilege to override the default home gateway when provisioning SIM-based IMEIs for this service. False: customer does not have the privilege to override the default home gateway when provisioning SIM-based IMEIs for this service.
demoAndTrialCounts	No	Array of demo and trial counts
demoAndTrialCount (type: datCountImpl)	No	
datType (type: datTypeEnum)	Yes	Demo or Trial type. See DAT Types.
allowedCount (type: int)	Yes	Allowed count for accounts of above type.
activeCount (type: int)	Yes	Current count of active accounts for type specified above.

BETA

Element/Field	Required?	Description
imeiSimLockPriority (type: imeiSimLockPriorityEnum)	Yes	See imeiSimLockPriorityEnum
prepaidDataEnabled (type:Boolean)	Yes	

BETA

13.26 findUserProfiles

The findUserProfile request is used to find the attributes of a user profile for a user account with SPNetPro access.

The findUserProfile request consists of the following pieces of data: (findUserProfilesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication. For this request calling user and password are required.
userId (type: string)	Yes	ID for user to filter results
userName (type: string)	Yes	Username for the user to be modified.
status (type: statusEnum)	No	User status. See User Account Status.
groupId (type: string)_	No	Provide group ID if desire is to filter by access group
includeInternal (type: Boolean)	Yes	Include internal users in result set

A typical response will contain the following (findUserProfilesResponseImpl)

Element/Field	Required?	Description
users (type: userImpl)	No	See User.

13.27 getAccountPoolingGroup

The getAccountPoolingGroup request is used to retrieve a specific pool group.

The getAccountPoolingGroup request consists of the following pieces of data: (getAccountPoolingGroupRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication. For this request calling user and password are required.
poolId (type: string)	Yes	Unique Id for the pool

BETA

Element/Field	Required?	Description
limit (type:int)	Yes	Number of records to be retrieved in a single call. The default limit is set to 5000 records.
startRow (type:int)	Yes	Start record that the query will be based off. This is used in combination with the “limit” field.

A typical response will contain the following (getAccountPoolingGroupResponseImpl)

Element/Field	Required?	Description
accountPoolingGroup (type: accountPoolingGroupImpl)	Yes	
id (type: string)	Yes	Id for the pool group.
status (type: statusEnum)	No	Current status of group. See Status.
name (type: string)	Yes	Name for the group (this will be unique within each service provider account)
planName (type: string)	No	Name of Pool group plan
planType (type: poolGroupTypeEnum)	No	Type of pool group
Description (type: string)	No	Description of the group attributes
totalNumberOfMemberAccounts (type: int)	No	Number of active accounts included on the group
memberAccounts	No	Array of member details
memberAccount (type: subscriberAccountEntryImpl)	No	Member account attributes
subscriberAccountNumber (type:string)	Yes	Contains the actual account number which represents the subscriber account. Follows the convention “SUB-XXXXXXX”.
requestType (type:string)	No	Type of request submitted
serviceType (type:string)	No	See Service Types.
atsInfo (type: atsInfoEntryImpl)	Yes	ATS related attributes
icao (type:string)	No	ICAO detail.

Element/Field	Required?	Description
airlineRegistration (type:string)	No	Airline registration id
channel (type: atsChannelTypeEnum)	No	ATS Channel type
subscriberOrganization (type:string)	No	Subscriber organization
deviceDetails (type:string)	No	Device attribute details
accountStatus (type:accountStatusEnum)	Yes	Contains the current status of the subscriber account. See Account Status.
statusDescription (type:string)	No	Description of status
createDate (type:string)	No	Date of activation
modifyDate (type:string)	No	Date of last modification
closedDate (type:string)	Yes	Date account was deactivated

13.28 getAccountPoolingGroups

The getAccountPoolingGroups request is used to retrieve a listing of pool groups available to the provider, based on service type.

The getAccountPoolingGroups request consists of the following pieces of data: (getAccountPoolingGroupsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication. For this request calling user and password are required.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
submarket (type: string)	No	Submarket for pool groups
poolGroupType (type: poolGroupTypeEnum)	No	See Pool Group Types.

BETA

Element/Field	Required?	Description
status (type: statusEnum)	No	Default ACTIVE. Current status of group. See Status.

A typical response will contain the following (getAccountPoolingGroupsResponseImpl)

Element/Field	Required?	Description
accountPoolingGroups	Yes	
accountPoolingGroup (type: accountPoolingGroupNoMembersImpl)	Yes	
spAccountNum (type: string)	Yes	Service Provider account number.
spAccountName (type: string)	Yes	Service Provider Name
id (type: string)	Yes	Id for the pool group.
status (type: statusEnum)	No	Current status of group. See Status.
name (type: string)	Yes	Name for the group (this will be unique within each service provider account)
serviceType (type: serviceTypeEnum)	No	Service Type of the group. See Service Types
planInfo (type: string)	No	Description of Group plan attributes.

BETA

13.29 getAirlines

The getAirlines request is used to retrieve the available airline names used for ATS accounts.

The getAirlines request consists of the following pieces of data: (getAirlinesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.

A typical response will contain the following. (getAirlinesResponseImpl)

Element/Field	Required?	Description
Airlines	Yes	An array of available airlines.
airline (type: airlinesEntryImpl)	No	Will be provided for getAirlinesResponse.
id (type: string)	Yes	ID of airlines

Element/Field	Required?	Description
name (type:string)	Yes	Name of airlines.
airlineGroupId(type:string)	No	Group identifier for airlines.

13.30 getBulkActionRequestReport

The getBulkActionReport request is used to retrieve the report of results from bulk action requests.

The getBulkActionReport request consists of the following pieces of data:
(getBulkActionReportRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
bulkRequestId (type: string)	Yes	Id for the bulk request.
requestType (type: string)	No	Provide type of request to filter results if desired.

A typical response will contain the following:(getBulkActionRequestReportResponse)

Element/Field	Required?	Description
response (type:searchAccountResponseImpl)	Yes	
subscriberAccounts	No	Array of subscriber details matching search criteria
subscriberAccount (type: subscriberAccountEntryImpl)	No	This object contains the subscriber account information
subscriberAccountNumber (type:string)	Yes	Contains the actual account number which represents the subscriber account. Follows the convention “SUB-XXXXXXX”.
requestType (type:string)	No	Type of request submitted
serviceType (type:string)	No	See Service Types.
atsInfo (type: atsInfoEntryImpl)	Yes	ATS related attributes
icao (type:string)	No	ICAO detail.
airlineRegistration (type:string)	No	Airline registration id
channel (type: atsChannelTypeEnum)	No	ATS Channel type

BETA

Element/Field	Required?	Description
subscriberOrganization (type:string)	No	Subscriber organization
deviceDetails (type:string)	No	Device attribute details
accountStatus (type:accountStatusEnum)	Yes	Contains the current status of the subscriber account. See Account Status.
statusDescription (type:string)	No	Description of status
createDate (type:string)	No	Date of activation
modifyDate (type:string)	No	Date of last modification
closedDate (type:string)	Yes	Date account was deactivated
configuredResultLimit (type: int)	No	Limit of results returned
totalResultCount (type: int)	No	Total results available

13.31 getBundle

The getBundle request is used to get any bundle based on its unique identifier.

The getBundle request consists of the following pieces of data: (getBundleRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
bundleId (type:string)	No	This is the ID that the system associates with a bundle.

A typical response will contain the following (getBundleResponseImpl).

Element/Field	Required?	Description
bundle (type:bundleInfoImpl or type:bulkBundleInfoImpl or type:datBundleInfoImpl)	No	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.

BETA

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getBundle

Errors or exceptions

Error	What it means/How to fix it
The bundle identified by ID [<bundleId>] does not exist.	The bundleId parameter does not refer to an existing bundle.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.32 getBurstBroadcastCoverageAreas

The getBurstBroadcastCoverageAreas request is used to get a list of valid values for the Burst broadcast coverage areas for use in subscribing to Burst service. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getBurstBroadcastCoverageAreas request consists of the following pieces of data: (getBurstBroadcastCoverageAreasRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.

A typical response will contain the following. (getBurstBroadcastCoverageAreasResponseImpl)

Element/Field	**Required?**	**Description**
bcaList	Yes	This is the list of broadcast coverage areas available. This could be empty.
bca (type: string)	No	Identifier for broadcast coverage area.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getBurstBroadcastCoverageAreas

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception.

BETA

Error	What it means/How to fix it
	Please contact Iridium Service Delivery so they can investigate.

13.33 getBurstDeviceBundles

The getBurstDeviceBundles request is used to get a list of valid values for the Burst device bundle field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getBurstDeviceBundles request consists of the following pieces of data: (getBurstDeviceBundlesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Valid plan types for this request include: burstDevicePlan:burstDevicePlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the Burst Device bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

BETA

A typical response will contain the following. (getBundlesResponseImpl)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty.

		bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
bundle(type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getBurstDeviceBundles

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.34 getBurstSubscriptionNames

The getBurstSubscriptionNames request is used to get a list of valid values for the Burst subscription names. Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.

The getBurstSubscriptionNamesrequest consists of the following pieces of data: (getBurstSubscriptionNamesRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.

A typical response will contain the following. (getBurstSubscriptionNamesResponseImpl)

Element/Field	**Required?**	**Description**
subscriptions	Yes	This is the list of subscriptions available. This could be empty.
subscription (type:string)	No	Subscription name

BETA

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getBurstSubscriptionNames

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.35 getCallSummaryReport

The getCallSummaryReport request has been deprecated and will no longer be enabled. Please use createOnDemandReport or createSchedReport methods from the IWS Report endpoint in place of this request.

13.36 getCommitmentPeriods

The getCommitmentPeriods request is used to get a list of valid values for the commitmentPeriodBundleId field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getCommitmentPeriods request consists of the following pieces of data: (getCommitmentPeriodsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Specify one Plan object, type depends on product desired. openPortPlan:openPortPlanImpl burstDevicePlan:burstDevicePlanImpl sbdPlan:sbdPlanImpl m2mPlan:m2MPlanImpl

BETA

Element/Field	Required?	Description
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the Commitment Period bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following (getBundlesResponseImpl)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
bundle(type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getCommitmentPeriods

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.

BETA

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.37 getDataBundles

The getDataBundles() request is used to get a list of valid values for the dataBundleId field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getDataBundles request consists of the following pieces of data: (getDataBundlesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Specify one Plan object, type depends on product desired. openPortPlan:openPortPlanImpl burstDevicePlan:burstDevicePlanImpl sbdPlan:sbdPlanImpl m2mPlan:m2MPlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the Data bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following. (getBundlesResponseImpl)

BETA

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
bundle(type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getDataBundles

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.38 getDataSpeeds

The getDataSpeeds request is used to get a list of valid values for the dataSpeedBundleId field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getDataSpeeds request consists of the following pieces of data: (getDataSpeedsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.

BETA

Element/Field	Required?	Description
Plan	Yes	See Plan. Specify one Plan object, type depends on product desired. openPortPlan:openPortPlanImpl burstDevicePlan:burstDevicePlanImpl sbdPlan:sbdPlanImpl m2mPlan:m2MPlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the Data Speed bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following. (getBundlesResponseImpl)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
bundle(type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getDataSpeeds

BETA

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.39 getDemoAndTrialBundles

The getDemoAndTrialBundles request is used to get a list of valid values for the demoAndTrial field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getDemoAndTrialBundles request consists of the following pieces of data: (getDemoAndTrialBundlesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Specify one Plan object, type depends on product desired. openPortPlan:openPortPlanImpl burstDevicePlan:burstDevicePlanImpl sbdPlan:sbdPlanImpl m2mPlan:m2MPlanImpl
fromBundleId (type:string)	No	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the demoAndTrial bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions. Will default to 0.

BETA

Element/Field	Required?	Description
forActivate (type:boolean)	No	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation. Will default to false.

A typical response will contain the following. (getBundlesResponseImpl)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
datBundle(type:datBundleInfoImpl)	No	See datBundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getDemoAndTrialBundles

BETA

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.40 getDemoAndTrialCounts

The getDemoAndTrialCounts request is used to get a list of valid values for the demoAndTrial field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getDemoAndTrialCounts request consists of the following pieces of data: (getDemoAndTrialCountsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceProviderProfileConfigTypeEnum)	Yes	See Service Provider Profile Types

A typical response will contain the following. (getDemoAndTrialCountsResponseImpl)

Element/Field	Required?	Description
demoAndTrialCounts	No	
demoAndTrialCount (type: availableDemoAndTrialCountsImpl)	No	
serviceType (type: serviceProviderProfileConfigTypeEnum)	Yes	See Service Provider Profile Types
datType (type: datTypeEnum)	Yes	
availableCount (type: int)	Yes	

BETA

activeCount (type: int)	Yes	

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getDemoAndTrialCounts

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.41 getDevices

The getDevices request is used to get one or more valid devices that can be passed into the activation or update request.

The getDevices request consists of the following pieces of data: (getDevicesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
deviceType (type:deviceTypeEnum)	Yes	See Device Types.
deviceStatusType (type:deviceStatusTypeNum)	Yes	See Device Status Types.
submarketId (type:string)	No	
numResults (type:int)	No	Because there can be a large number of devices assigned to a service provider, this value can be set to limit the number of results returned. Default value is 0 which will return all devices that match criteria provided.
ip (type:deviceIpSearchParametersImpl) num (type:deviceNumSearchParametersImpl)	Yes	Each of these objects represents a different type of search criteria. The IP, Num, and Apn search parameter objects are provided to allow for additional filtering of results.

BETA

Element/Field	Required?	Description
Apn (type: deviceApnSearchParametersImpl) Each type will be outlined below.		

The device search parameters should be used to locate VPN and SIM devices. It can also be used to search for other device types if additional filtering is not required:

The device IP search parameters should be used to locate IP devices:

Element/Field	Required?	Description
ip (type:deviceIpSearchParametersImpl)	Yes	
apnId (type:string)	No	This allows the results to be narrowed/filtered by the VPN. To do so, this should be set to the ID of the VPN object. See VPN.

The device NUM search parameters should be used to locate Num devices:

Element/Field	Required?	Description
num (type:deviceNumSearchParametersImpl)	Yes	
numberType (type: deviceNumberTypeEnum)	Yes	This allows the results to be narrowed/filtered by the type of number.
networkElement (type:string)	No	This allows the results to be narrowed/filtered by the network. This is useful when trying to find correspondingSIMs and MSISDNs. See SIM and Number for their related network fields.

The device APN search parameters should be used to locate Data network (VPN/APN) devices:

BETA

Element/Field	Required?	Description
apn (type: deviceApnSearchParametersImpl)	Yes	
apnId (type: xs:string)	No	
coloTypes	No	Array of coloTypes
collotype (type: ColoDataNWOptionsEnum)	No	This allows the results to be narrowed/filtered by the type of number. See coloDataNWOptionsEnum
transitionAPN (type:string)	No	ID of existing APN. When provided will return requested APN and all other APN that match search criteria. Used to populate list if existing contract APN has been made obsolete

A typical response will contain the following. (getDevicesResponseImpl)

Element/Field	Required?	Description
Devices	Yes	This is the list of results that matches the search criteria.
device (type:deviceImpl) vpn (type:deviceApnImpl) sim (type:deviceSimImpl) num (type:deviceNumberImpl) ip (type:deviceIpImpl) capcode (type: deviceCapcodeImpl)	No	See SIM, VPN, IP, Number, Capcode.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getDevices

BETA

Errors or exceptions

Error	What it means/How to fix it
No search parameters were provided.	The request contained no device search parameters.
Device type was null.	The deviceType value within the search parameter was empty or invalid.
Service Provider must be specified.	The service provider associated with account number XXXXXX could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

1.1. getDIDWWNumberInventory

The getDIDWWNumberInventory request is used to pull DIDWW numbers from the Iridium billing system. The request consists of the following pieces of data. (getDIDWWNumberInventoryRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
localNumberCountryId (type: string)	No	See getLocalNumberRegions
status (type: deviceStatusTypeEnum)	No	See deviceStatusTypeEnum

If the request is successful IWS will return a response object with a list of local number countries and their local numbers. (getDIDWWNumberInventoryResponseImpl)

Element/Field	Required?	Description
localCountry (type: string)	Yes	An array of local number countries
localCountry (type: didwwLocalCountryImpl)	No	
id (type:string)	No	Local country Id
name (type:string)	No	Name of the
deviceModel (type:string)	No	Model of the country. Format PLUS##
devices	No	An array of local number devices

BETA

Element/Field	Required?	Description
device (type: DeviceNumberImpl)	No	
createdDate (type:createdDate)	No	An array of devices devices that failed to populate in the billing system.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=getDIDWWNumberInventory

13.42 getEmailNotification

The getEmailNotification request will allow the service provider to view the current configuration for email notifications. The report provides a detail of which action types will result in email notification being sent to the service provider email account.

The getEmailNotification request consists of the following pieces of data

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.

A typical response will contain the following. (getEmailNotificationResponseImpl)

Element/Field	Required?	Description
email	Yes	
notifications (type: actionNotificationImpl)	No	List of notification name and enabled options
action (type: actionNotificationImpl)	Yes	Array of actions
actionType (type: emailNotificationTypeEnum)	Yes	Action type id. See Notification Action Types
description (type: string)	Yes	Display name of action type
name (type: string)	Yes	Name of notification option
enabled (type: string)	Yes	Is the notification option enabled? TRUE – Enabled FALSE - Disabled

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getEmailNotification

BETA

13.43 getEvoucherInventory

The evoucher inventory allows a service provider to view inventory of telephony eVouchers available to their account. The report provides a summary of the number of units, minutes, and quantity of vouchers available for each voucher type.

The getEVoucherInventory request consists of the following pieces of data

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
limit (type: int)	Yes	Number of records to be retrieved in a single call. Default: Enter 0 for no max limit
startRow (type: int)	Yes	Start row record that the query will be based off. This is used in combination with the "limit" field. Default: Enter 1 for starting row

A typical response will contain the following. (getEVoucherInventoryResponseImpl)

Element/Field	Required?	Description
totalNumberOfRecords (type: int)	No	Total count of records returned.
evoucherInventoryDetails	Yes	Array of evoucher inventory search criteria. Can be empty.
evoucher InventoryDetail (type:evoucherInventoryEntryImpl)	No	The EVoucher Inventory object item.
voucherType (type: string)	Yes	The code identifying the type of e-voucher.
unitDenomination (type: string)	Yes	The number of units on each e-voucher.
minDenomination (type: string)	Yes	The number of minutes provided by each eVoucher.
quantityAvailable (type: string)	Yes	The number of e-vouchers of the type and denomination.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getEvoucherInventory

DRAFT

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>]	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error gathering EVoucher Inventory	If this error is returned, there was a problem with the Evoucher Inventory IWS call. Please contact Iridium Service Delivery so they can investigate.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.44 getIwsRequest

The getIwsRequest call is used to pull detailed information about specified IWS request. NOTE: IWS records are temporary and will be purged after <timeframe>

The getIwsRequest call consists of the following pieces of data

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
requestId (type: string)	Yes	ID value of specific IWS request.
iwsLogType (type: iwsLogTypeEnum)	Yes	See IWS Log Type

A typical response will contain the following. (iwsLogDetailResponseImpl)

Element/Field	Required?	Description
iwsLog (type: iwsLogImpl)	No	Object with detailed IWS Request information
requestId (type: string)	Yes	This is the ID that the system uses to identify the Request
requestUser (type: string)	Yes	IWSUsername used in the IWS request
requestedOperation (type: string)	Yes	Operation of the request

caller (type: string)	No	Caller value in the request
requestedTs (type: string)	Yes	Timestamp the request was made
errorMessage (type: string)	No	Provides the exception handling error that was returned in response to request if the request was in error
request (type: string)	No	Request in XML format written with escape string
response (type: string)	No	Response in XML format written with escape string

13.45 getIwsRequests

The getIwsRequests call is used to pull a list of IWS request made by users within a timeframe. NOTE: IWS records are temporary and will be purged after <timeframe>

The getIwsRequests call consists of the following pieces of data

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
iwsOperation (type: string)	Yes	ID value of specific IWS request.
startDate (type: string)	No	Search start time for filtering IWS request. All timestamps in IWS follow the ISO 8601 format.
endDate (type: string)	No	Search end time for filtering IWS request. All timestamps in IWS follow the ISO 8601 format.
Limit (type: int)	No	Search limit
startRow (type: int)	No	Search start row of returned results

A typical response will contain the following. (iwsLogDetailResponseImpl)

Element/Field	Required?	Description
iwsLog (type: iwsLogImpl)	No	Object with detailed IWS Request information
requestId (type: string)	Yes	This is the ID that the system uses to identify the Request
requestUser (type: string)	Yes	IWSUsername used in the IWS request
requestedOperation (type: string)	Yes	Operation of the request

BETA

caller (type: string)	No	Caller value in the request
requestedTs (type: string)	Yes	Timestamp the request was made
errorMessage (type: string)	No	Provides the exception handling error that was returned in response to request if the request was in error
request (type: string)	No	Request in XML format written with escape string
response (type: string)	No	Response in XML format written with escape string

13.46 getLanguages

The getLanguages request is used to get a list of valid values for the languageBundleId field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getLanguages request consists of the following pieces of data: (getLanguagesRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Specify one Plan object, type depends on product desired. openPortPlan:openPortPlanImpl burstDevicePlan:burstDevicePlanImpl sbdPlan:sbdPlanImpl m2mPlan:m2MPlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the Language bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not.

BETA

Element/Field	Required?	Description
		false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following. (getBundlesResponseImpl)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
bundle(type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getLanguages

BETA

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.47 getLanguageChoices

The getLanguageChoices request is used to get a list of valid values for the language field.

The getLanguageChoices request consists of the following pieces of data: (getLanguageChoicesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceLanguageType (type: serviceLanguageTypeEnum)	No	Type of service to filter language choices.

A typical response will contain the following. (getLanguageChoicesResponseImpl)

Element/Field	Required?	Description
languageChoices	Yes	Array of language choices
languageChoice (type: languageChoiceImpl)	No	Language object
Id (type string)	Yes	Id for language
serviceLanguageType	Yes	Service type for language choice
languageName	Yes	Language display name

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getLanguageChoices

BETA

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.48 getLocalNumberRegions

The getLocalNumberRegionsrequest is used to get a list of valid regions offering local numbers (singlestage dialing) for the selected plan.

The getLocalNumberRegionsrequest consists of the following pieces of data: (getLocalNumberRegionsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
servicePlanId	Yes	Plan to be used with local numbers

A typical response will contain the following. (getLocalNumberRegionsResponseImpl)

Element/Field	Required?	Description
localNumberCountries	Yes	Array of local number region choices
localNumberCountry (type: localNumberCountryImpl)	No	Local Number Region object
id (type: string)	Yes	Id for local number type
name (type: string)	Yes	Local Number Region display name

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getLocalNumberRegions

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.49 getM2MBundles

The getM2MBundles request is used to get a list of valid values for the m2mBundleId field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getM2MBundles request consists of the following pieces of data: (getM2MBundlesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Valid plan types for this request include: m2mPlan:m2mPlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the M2M bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following.

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl

BETA

Element/Field	Required?	Description
		See Bundles.
bundle(type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getM2MBundles

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.50 getManufacturers

The getManufacturers request is used to get a list of valid manufacturers.

The getManufacturers request consists of the following pieces of data: (getManufacturers RequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
status (type: objectStatusEnum)	No	See Object Status.

A typical response will contain the following. (getManufacturersResponseImpl)

Element/Field	Required?	Description
manufacturers	Yes	Array of manufacturers
manufacturer(type: manufacturerImpl)	No	Manufacturer object
id (type: string)	No	
name (type: string)	No	
verticals	No	Array of supported verticals
vertical (type:string)	No	Name of vertical supported
status (type: objectStatusEnum)	No	Status of the manufacturer object ACTIVE – manufacturer is valid

BETA

Element/Field	Required?	Description
		OBSOLETE – manufacturer is no longer supported.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getManufacturers

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.51 getModels

The getModels request is used to get a list of valid device models.

The getModels request consists of the following pieces of data: (getModelsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
manufacturerId(type:string)	No	Provide a manufacturerId if it is desired to filter results by manufacturer. See getManufacturers call for details about retrieving valid ids.
name (type:string)	No	Provide a model name if it is desired to filter results by name. Wildcard values supported
modelNumber (type: string)	No	Provide a model number if it is desired to filter results by model number. Wildcard values supported
verticalId (type: string)	No	Provide a vertical Id if it is desired to filter results by vertical. See getVerticals call for details about retrieving valid ids.
status (type: objectStatusEnum)	No	See Object Status.

A typical response will contain the following. (getModelsResponseImpl)

BETA

Element/Field	Required?	Description
models (type: getModelsResponseImpl)	Yes	Array of models
model (type: simplifiedModelImpl)	No	Model object
id (type: string)	No	
vertical (type: string)	No	
manufacturerId(type:string)	No	The ID for the manufacturer linked to the model.
name (type:string)	No	The name of the model as provided by the manufacturer
modelNumber (type: string)	No	The model Number as provided by the manufacturer.
transceiverModel (type:string)	No	The model designation for the transceiver embedded in the device.
transceiverType (type: transceiverTypeEnum)	No	See transceiverTypeEnum
postpaidCapable (type: string)	No	Model is postpaid capable
prepaidDataCapable (type: string)	No	Model is prepaid capable
deviceMgrCapable (type: string)	No	Model device manager will be enabled or not
sdfCapable (type:string)	No	Model allows SDFs on Certus contracts
status (type: objectStatusEnum)	No	Status of the model object ACTIVE – model is valid for new provisioning requests OBSOLETE – model is no longer supported for new provisioning requests.
certifiedFlag (type: string)	No	Indicates if the model is certified for provisioning requests true – model is valid for new provisioning requests false – model is not supported for new provisioning requests.
seClassMark (type: string)	No	Configuration parameter for class mark attributes

BETA

Element/Field	Required?	Description
maxDataSpd (type: maxDataBandwidthEnum)	No	See maxDataBandwidthEnum
modCod (type: int)	No	Configuration parameter for c8 attributes
imeiSimBasedFlag (type:string)	No	true – this model can support a sim false – this model cannot support a sim
coloDataNWOptions (type: coloDataNWOptionsEnum)	No	See coloDataNWOptionsEnum
voicePrepaidOptions (type: voicePrepaidOptionsEnum)	No	See voicePrepaidOptionsEnum
maxVoiceLines (type:int)	No	Determines how many voices lines a model can support
voiceQuality (type: voiceQualityEnum)	No	Defines the quality of voice services that this model can support.
dataPrepaidOptions (type: dataPrepaidOptionsEnum)	No	Defines the types of Prepaid Data services that this model can support.
maxSdfAllowed (type:int)	No	Determines how many secondary data flows a model can support

BETA

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getModels

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.52 getModel

The getModel request is used to get the full details about a specific model.

The getModel request consists of the following pieces of data: (getModelRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.

Element/Field	Required?	Description
modelId(type:string)	Yes	Provide the modelId number for the requested model. See getModels method for valid values.

A typical response will contain the following. (getModelResponseImpl)

Element/Field	Required?	Description
model (type: modelImpl)	No	Model object
id (type: string)	No	
vertical (type: string)	No	
manufacturerId(type:string)	No	The ID for the manufacturer linked to the model.
name (type:string)	No	The name of the model as provided by the manufacturer
modelNumber (type: string)	No	The model Number as provided by the manufacturer.
transceiverModel (type:string)	No	The model designation for the transceiver embedded in the device.
transceiverType (type: transceiverTypeEnum)	No	See transceiverTypeEnum
postpaidCapable (type:string)	No	Model is postpaid capable
prepaidDataCapable (type:string)	No	Model is prepaid capable
deviceMgrCapable (type:string)	No	Model device manager will be enabled or not
sdfCapable (type:string)	No	Model allows SDFs on Certus contracts
geocastCapable (type:string)	No	
sbdCapable (type: string)	No	true – this model can support SBD services false – this model cannot support SBD services
pttCapable (type: string)	No	true – this model can support PTT services false – this model cannot support PTT services

BETA

Element/Field	Required?	Description
burstCapable (type: string)	No	true – this model can support Burst services false – this model cannot support Burst services
opCapable (type: string)	No	true – this model can support OpenPort services false – this model cannot support OpenPort services
telCapable (type: string)	No	true – this model can support Telephony services false – this model cannot support Telephony services
certusVoiceCapable (type: string)	No	true – this model can support Certus Voice services false – this model cannot support Certus Voice services
maxSdf (type: int)	No	Determines how many secondary data flows a model can support
voiceQuality (type: voiceQualityEnum)		See voiceQualityEnum
maxVoiceLines (type:int)		Determines how many voices lines a model can support
streamingCapable (type: string)	No	true – this model can support Certus Streaming services false – this model cannot support Certus Streaming services
maxStreamingSpd (type: streamingSpeedEnum)	No	Indicates maximum streaming speed supported by the model.
streamingMaxConcurrentSessions (type:int)	No	Max number of streaming sessions permitted
streamingMaxAggrRate (type:int)	No	Max rate across all concurrent sessions
maxDataSpeed (type: maxDataBandwidthEnum)	No	Indicates maximum data speed supported by the model.
serviceCat (type: modelServiceCategoryEnum)	No	Service category of the model See Service Categories

BETA

Element/Field	Required?	Description
serviceClass (type: modelServiceClassEnum)	No	Service class of the model See Service Classes
classmarks	No	An array of classmarks
classmark (type: dataClassmarkEnum)	No	Classmark of the model See Data Classmark
coloDataNWOptions(type: coloDataNWOptionsEnum)	No	See coloDataNWOptionsEnum
dataPrepaidOptions (type: dataPrepaidOptionsEnum	No	See dataPrepaidOptionsEnum
voicePrepaidOptions (type: voicePrepaidOptionsEnum)	No	See voicePrepaidOptionsEnum
status (type: objectStatusEnum)	No	Status of the model object ACTIVE – model is valid for new provisioning requests OBSOLETE – model is no longer supported for new provisioning requests.
certT (type: string)	No	Timestamp for certification
certBy (type: string)	No	User which certified the model
safetyCapable (type: string)	No	true – this model can support Safety services false – this model cannot support Safety services
maxB1Spd (type: int)	No	Max speed under Block 1 Satellite
modCod (type: int)	No	
imeiSimBasedFlag (type:string)	No	true – this model can support a sim false – this model cannot support a sim

BETA

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getModel

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.53 getPrepaidAccountDetail

The prepaid account detail shows the view account details of a particular prepaid account and unit expiration notice. It can be searched by MSISDN.

The getPrepaidAccountDetail request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
msisdn (type: string)	Yes	Represents the phone number for VOICE calls. MSISDN-A is only used for prepaid.

A typical response will contain the following. (GetPrepaidAccountDetailResponseImpl)

Element/Field	Required?	Description
prepaidAccountDetail (type: PrepaidAccountDetailImpl)	Yes	The prepaid account detail object item. See Prepaid Account Detail.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getPrepaidAccountDetail

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error gathering Prepaid Account Detail	If this error is returned, there was a problem with the Prepaid Account Detail

BETA

Error	What it means/How to fix it
	IWS call. Please contact Iridium Service Delivery so they can investigate.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.54 getPrepaidAccountSummary

The prepaid account summary displays the account details of each prepaid account owned by the service provider, it can be limited by plan, create date, expiration date, balance, or any combination of those.

The getPrepaidAccountSummary request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
prepaidAccountType (type: prepaidAccountTypeEnum)	Yes	See prepaidAccountTypeEnum
subscriberitem (type: string)	Yes	The request subscriber item for search can either be an MSISDN or SIM
servicePlan (type: string)	No	Prepaid plan selection value to find accounts.
createdDateStart (type: string)	No	Starting creation date value for search. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
createdDateEnd (type: string)	No	Ending creation date value for search. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
expiryDateStart (type:string)	No	Starting expiration date value for search. (Valid only for Prepaid account type) Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
expiryDateEnd (type:string)	No	Ending expiration date value for search. (Valid only for Prepaid account type) Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
balance (type: string)	No	Prepaid account balance value of available units. (Valid only for Prepaid account type)
balanceOperator (type: string)	No	Balance indicator to identify search for accounts with balance (<, <=, >, >=, or =) to value provided. (Valid only for Prepaid account type)

BETA

Element/Field	Required?	Description
limit (type: int)	Yes	Number of records to be retrieved in a single call. Default: Enter 0 for no max limit
startRow (type: int)	Yes	Start row record that the query will be based off. This is used in combination with the “limit” field. Default: Enter 1 for starting row

A typical response will contain the following. (GetPrepaidAccountSummaryResponseImpl)

Element/Field	Required?	Description
totalNumberOfRecords (type: int)	No	Total count of records returned.
prepaidAccountSummaryDetails	Yes	Array of prepaid account summary search criteria. Can be empty.
prepaidAccountSummaryDetail (type: PrepaidAccountDetailImpl)	No	The prepaid account detail object item. See Prepaid Account Detail

To generate sample requests and responses:

https://training.iridium.com/iwsexplorer/index.php?call=getPrepaidAccountSummary

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>]	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error Gathering Prepaid Plan Bundles	If this error is returned, there was a problem with the prepaid plan IWS call. Please contact Iridium Service Delivery so they can investigate.
Error on starting/ending created date	If this error is returned, there was a problem with the date formatting or no date string was entered. Check if the date string is formatted with the following string format: yyyy-MM-dd'T'HH:mm:ss'Z'

BETA

Error	What it means/How to fix it
Error on starting/ending expiration date	If this error is returned, there was a problem with the date formatting or no date string was entered. Check if the date string is formatted with the following string format: yyyy-MM-dd'T'HH:mm:ss'Z'
Error gathering Prepaid Account Summary	If this error is returned, there was a problem with the Prepaid Account Summary IWS call. Please contact Iridium Service Delivery so they can investigate.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.55 getPrepaidPlans

The getPrepaidPlans request is used to get a listing of the Prepaid/Crew plans returned by prepaid account type and submarket. The getPrepaidPlans request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
prepaidAccountType (type:prepaidAccountTypeEnum)	Yes	See PrepaidAccountTypeEnum.
submarketId (type:string)	No	

A typical response will contain the following. (getPrepaidPlansResponseImpl)

Element/Field	Required?	Description
submarkets	No	This is a list of submarkets.
submarket (type:submarketPrepaidPlansImpl)	No	
entry (type:string)	No	
Prepaid	No	This is a list of prepaid plans.
plan (type:string)	No	
Crew	No	This is a list of crew plans.
plan (type:string)	No	This is the crew plan available.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getPrepaidPlans

BETA

13.56 getProgrammableButtons

The getProgrammableButtons request is used to get a list of valid values for the programmable buttons fields in the Number object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getProgrammableButtons request consists of the following pieces of data: (getProgrammableButtonsRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.

A typical response will contain the following. (getProgrammableButtonsResponseImpl)

Element/Field	**Required?**	**Description**
programmableButtons	Yes	This is the list of programmable buttons available. This could be empty.
programmableButton (type: programmableButtonImpl)	No	
code (type: string)	Yes	Short Code to be programmed for button
name (type: string)	Yes	Name of button value

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getProgrammableButtons

Errors or exceptions

Error	**What it means/How to fix it**
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.57 getProjectVersion

The getProjectVersion request is used to display the current version of IWS.

The getProjectVersion request consists of the following pieces of data: (getProjectVersionRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.

A typical response will contain the following. (getProjectVersionResponseImpl)

Element/Field	Required?	Description
Version (type:string)	Yes	Current version of the project deployment

13.58 getPromoBundles

The getPromoBundles request is used to get a list of valid values for the promo field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getPromoBundles request consists of the following pieces of data: (getPromoBundlesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Specify one Plan object, type depends on product desired. openPortPlan:openPortPlanImpl burstDevicePlan:burstDevicePlanImpl sbdPlan:sbdPlanImpl m2mPlan:m2MPlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the promo bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.

BETA

Element/Field	Required?	Description
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following. (getBundlesResponseImpl)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
Bundle(type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getPromoBundles

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.59 getQueueEntries

The getQueueEntries request is used to get a list of requests in the background processing queue..

The getQueueEntries request consists of the following pieces of data: (getQueueEntriesRequestImpl)

BETA

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
endpoint (type: endpointEnum)	Yes	Origination IWS endpoint for the request
status (type: queueEntryStatusEnum)	Yes	Status of the request
createdBy (type: string)	Yes	User submitting request

A typical response will contain the following. (getQueueEntriesResponseImpl)

Element/Field	Required?	Description
queueEntries	Yes	The list of queue entries available
queueEntry (type: baseQueueEntryImpl)	No	
id (type: string)	Yes	Request ID, unique value
operation (type: string)	Yes	IWS Operation for the request
endpoint (type: endpointEnum)	Yes	Origination IWS endpoint for the request
status (type: queueEntryStatusEnum)	Yes	Status of the request
errorMessage (type: string)	No	Error message from the response if the request was not successful
createdBy (type: string)	Yes	User submitting request
serviceProviderNumber (type: string)	Yes	SP Account Number
serviceProviderName (type: string)	Yes	Service Provider Name

BETA

13.60 getQueueEntry

The getQueueEntry request is used to get a list of queued requests and their current status.

The getQueueEntry request consists of the following pieces of data: (getQueueEntryRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
queueEntryId (type: string)	Yes	Entry reference ID

A typical response will contain the following. (getQueueEntryResponseImpl)

Element/Field	Required?	Description
baseQueueEntryImpl	Yes	
baseQueueEntryImpl	Yes	Attributes for the queue request
id (type: string)	Yes	Request ID, unique value
operation (type: string)	Yes	IWS Operation for the request
endpoint (type: endpointEnum)	Yes	Origination IWS endpoint for the request
status (type: queueEntryStatusEnum)	Yes	Status of the request
errorMessage (type: string)	No	Error message from the response if the request was not successful
createdBy (type: string)	Yes	User submitting request
serviceProviderNumber (type: string)	Yes	SP Account Number
serviceProviderName (type: string)	Yes	Service Provider Name
serviceType (type: string)	No	Service type related to the request
request (type: string)	Yes	Raw IWS request XML
response (type: string)	No	Raw IWS response XML

13.61 getRechargeHistory

This recharge history report displays the vouchers that have been associated to a SIM prepaid account.

The getRechargeHistory request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
msisdn (type: string)	Yes	Represents the phone number for VOICE calls. MSISDN-A is only used for prepaid.
iccid (type: string)	Yes	This is the number representing a SIM card. Here are some examples: 8988169703000007348 8988169777000016038 The bold numbers in the middle of the SIM number represent the network that the device belongs to. This same series of numbers is present in the MSISDNs that are activated on the account. When not applicable, it will be populated with "NA".
limit (type: int)	Yes	Number of records to be retrieved in a single call. Default: Enter 0 for no max limit
startRow (type: int)	Yes	Start row record that the query will be based off. This is used in combination with the "limit" field. Default: Enter 1 for starting row

A typical response will contain the following. (GetRechargeHistoryResponseImpl)

Element/Field	Required?	Description
totalNumberOfRecords (type: int)	Yes	Total count of records returned.
rechargeHistoryDetails	Yes	Array of recharge history objects. Can be empty.
rechargeHistoryDetail (type: rechargeHistoryEntryImpl)	Yes	The recharge history object item.
account Number (type: string)	Yes	Account Number for search result.
voucherDate (type: string)	Yes	Date and time (GMT) of the voucher activation

BETA

Element/Field	Required?	Description
		Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
serialNumber (type: string)	Yes	The e-voucher serial number
voucherType (type: string)	Yes	The distribution type of e-voucher.
addedUnitsOrDays (type: string)	Yes	The added number of units for current Balance. Units: 30, 3000, 30000
oldBalance (type: string)	Yes	The old number of units Units: 15, 1500, 150000
newBalance (type: string)	Yes	The new number of units for voucher after recharge with units. Example: The old unit balance of 100 added with 500 units will change the unit balance to the new unit balance of 600 units.
oldExpirationDate (type: string)	Yes	Original date and time (GMT) of the voucher expiry prior to the recharge request. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' If the recharge request is for time units
newExpirationDate(type: string)	Yes	The new expiration date after the recharge request. Date and time (GMT) of the voucher expiry. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
createdBy (type: string)	Yes	Prepaid recharge activity user that created account. Same as username
rechargeFlag (type: boolean)	Yes	Shows whether the voucher has been created or recharged false – CREATE

BETA

Element/Field	Required?	Description
		true - RECHARGE
isRefunded (type: boolean)	Yes	Shows whether the current voucher has been refunded

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getRechargeHistory

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error gathering Recharge History	If this error is returned, there was a problem with the Recharge History IWS call. Please contact Iridium Service Delivery so they can investigate.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.62 getRudicsGroups

The getRudicsGroups request is used to get a list of valid values for the RUDICS Group field in the [Plan](#) object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getRudicsGroups request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See [Authentication](#).
params (type: rudicsGroupSearchParametersImpl)	Yes	
serviceType (type: servicetypeEnum)	Yes	See [Service Types](#).

A typical response will contain the following. (getRudicsGroupsResponseImpl)

Element/Field	Required?	Description
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
rudicsGroups	Yes	
rudicsGroup (type: rudicsGroupImpl)	No	This is the list of RUDICS Groups available. This could be empty.
id (type: string)	Yes	Internal system identifier for the group. This is to be used when adding a RUDICS group to an activation request.
name (type: string)	Yes	Name of the group
description (type: string)	Yes	**Description** for the group
group (type: string)	Yes	Internal billing system Group ID

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getRudicsGroups

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.63 getSBDBundles

The getSBDBundles request is used to get a list of valid values for the sbdBundleId field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getSBDBundles request consists of the following pieces of data: (getSBDBundlesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Valid plan types for this request include: sbdPlan:sbdPlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0.

BETA

Element/Field	Required?	Description
		Otherwise, it should be set to the value of the SBD bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following. (getBundlesResponseImpl)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
bundle(type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getSBDBundles

BETA

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.64 getScratchCardActivationStatus

The getScratchCardActivationStatus request consists of the following pieces of data. (searchScratchCardActivationStatusRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
searchType (type: scratchCardActivationStatusSearchTypeEnum)	Yes	See Scratch Card Activation Search Types.
value (type:string)	Yes	Either the purchase order number or the request ID, depending on the search type.

If the request is successful, a response will be returned with the following data. (searchScratchCardActivationStatusResponseImpl)

Element/Field	Required?	Description
activationList	No	Array of status objects
activationStatus (type:scratchCardActivationStatusImpl)	No	See Scratch Card Activation Status.
requestId (type:string)	Yes	Request identifier
purchaseOrder (type:string)	No	PO Number
status (type:activationStatusEnum)	Yes	See Activation Status.
type (type:string)	Yes	See Activation Type.
startSerial (type:string)	Yes	The first serial number in the batch.

BETA

Element/Field	Required?	Description
endSerial (type:string)	Yes	The final serial number in the batch.
requestTimestamp (type: string)	No	The time that the activation request was made.
totalRequested (type:int)	No	The number of scratch cards that were requested to be activated.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=getScratchCardActivationStatus

Errors or exceptions

Error	What it means/How to fix it
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.65 getScratchCardBatches

This report is a summary of available scratch cards, based on voucher type and batch number. The batch number is a unique identifier for a particular batch of scratch cards, usually referencing a purchase order. The voucher type is the distribution type, this indicates if the voucher is a virtual card (not physical card supplied by iridium) or a physical card, which iridium prints and sells.

The getScratchCardBatches request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: scratchCardServiceTypeEnum)	Yes	Filter by service type See Scratchcard Service Types.
limit (type: int)	Yes	Number of records to be retrieved in a single call. Default: Enter 0 for no max limit
startRow (type: int)	Yes	Start row record that the query will be based off. This is used in combination with the “limit” field. Default: Enter 1 for starting row

A typical response will contain the following. (GetScratchCardBatchesResponseImpl)

BETA

Element/Field	Required?	Description
totalNumberOfRecords (type: int)	No	Total count of records returned. Enter 0 if no records are returned.
scratchCardBatches	Yes	Array of scratchcard batch items. Can be empty.
scratchCardBatchDetail (type: scratchCardBatchesEntryImpl)	No	Scratch card batch object item that stores detailed data.
voucherType (type: string)	Yes	Distribution type that can represent physical or virtual scratch cards. Examples: - CREWRETAILPHYS and CREWRETAILVIRT are legacy scratch cards that are no longer produced.GOCHATPHYS and GOCHATVIRT are the replacement cards
batchNumber (type: string)	Yes	This is a unique identifier assigned by service delivery, usually containing the purchase order number
totalBatch (type: int)	Yes	This is the total number of vouchers in the that batch, identified by the batch number
availableToActivate (type: int)	Yes	Number of vouchers in the batch that are not activated.
activeForUse (type: int)	Yes	These are scratch cards that have been activated by the SP and ready to use, but haven’t had the first call to be considered in use.
inUse (type: string)	Yes	These are scratch cards that have been activated, and have had at least one call to be considered in use
expired (type: int)	Yes	These are cards that were in use, but a year has passed and they are now expired and may not make calls
usedUp (type: int)	Yes	These cards that have used all of their allotted minutes
startSerialNumber (type: string)	Yes	Start serial number for batch

BETA

Element/Field	Required?	Description
endSerialNumber (type: string)	Yes	End serial number for batch.

To generate sample requests and responses: https://training.iridium.com/iwsexplorer/index.php?call=getScratchCardBatches

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error gathering Scratch Card Batches	If this error is returned, there was a problem with the Scratch Card Batches IWS call. Please contact Iridium Service Delivery so they can investigate
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.66 getScratchcardDeals

The getScratchcardDeals call will pull a list of deals used for generating scratchcard batches based on submarket. The getScratchcardDeals request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
submarket (type: xs:string)	Yes	Filter by service type See Scratchcard Service Types.

A typical response will contain the following. (getScratchcardDealsResponseImpl)

Element/Field	Required?	Description
totalNumberOfRecords (type: int)	No	Total count of records returned. Enter 0 if no records are returned.
deals	No	Array of deals. Can be empty.

Element/Field	Required?	Description
CertusScratchcardDeal (type: CertusScratchcardDealImpl)	No	The scratchcards deal object item.
id (type: long)	Yes	Scratchcard ID
name (type: string)	Yes	Scratchcard name
obsolete (type: boolean)	Yes	The deal is obsolete and cannot be used for new request
networkPlan (type: string)	Yes	Network identifier
denomination (type:int)	Yes	Included data value with scratchcards with this deal. Quantity in MB.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getScratchcardDeals

13.67 getScratchcardOrders

The getScratchcardOrders call returns a list of existing orders and the list of currently active cards on the order. The getScratchcardOrders request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
orderId (type: string)	No	Filter for order id
searchStartDate (type: string)	No	If searching by a date range, this is the starting date of that range.
searchEndDate (type: string)	No	If searching by a date range, this is the ending date of that range.
serialNumber (type: string)	No	Filter for Serial number. Order pulled that contains this card.

A typical response will contain the following. (getScratchcardOrdersResponseImpl)

Element/Field	Required?	Description
orders	No	Array of deals. Can be empty.
CertusScratchcardOrder (type: certusScratchcardOrderImpl)	No	See Certus Scratchcard Order

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getScratchcardOrders

BETA

13.68 getScratchcards

The Scratchcards report allows SPs to query details on their scratchcards based on a set of customized set of inputs which include batch number, serial, balance, value, and dates.

The getScratchCards request consists of the following pieces of data:

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
serviceType (type: scratchCardServiceTypeEnum)	Yes	Filter by service type See Scratchcard Service Types.
searchType1 (type: telephonyScratchCardSearchTypeEnum)	Yes	See Scratchcards Search Types.
searchOperator1 (type: searchOperatorEnum)	Yes	Query operator associated with searchType1 item to search on Options: =, < , <=, >, >=, LIKE
searchValue1 (type: string)	Yes	Query value associated with searchType1 item to search on
searchType2 (type: telephonyScratchCardSearchTypeEnum)	No	See Scratchcards Search Types.
searchOperator2 (type: searchOperatorEnum)	No	Query operator associated with searchType2 item to search on Options: =, < , <=, >, >=, LIKE
searchValue2 (type: string)	No	Query value associated with searchType2 item to search on
realtime(type: Boolean)	Yes	TRUE: Will get realtime balance and status information, search criteria type must be SERIAL_NUMBER, with one single card value to search. FALSE: Use datamart report source and retrieve more than one voucher at a time.
limit (type: int)	Yes	Number of records to be retrieved in a single call. Default: Enter 0 for no max limit

BETA

Element/Field	Required?	Description
startRow (type: int)	Yes	Start row record that the query will be based off. This is used in combination with the “limit” field. Default: Enter 1 for starting row

A typical response will contain the following. (getScratchCardsResponseImpl)

Element/Field	Required?	Description
totalNumberOfRecords (type: int)	No	Total count of records returned. Enter 0 if no records are returned.
scratchCards	Yes	Array of scratch cards. Can be empty.
scratchCardsDetail (type: scratchCardsDetailEntryImpl)	No	The scratchcards detail object item.
serialNumber (type: string)	Yes	Serial number of scratchcards detail.
distributionType (type: string)	Yes	The distribution type of the scratchcards account.
activationDate (type: string)	Yes	The activation date for scratchcards account. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
expirationDate (type: string)	Yes	The expiration date for the scratchcards account.
balance (type:int)	Yes	The remaining balance on the scratchcards account.
servicePlan (type: string)	Yes	Service plan of scratch card account. Type of line used to make the call.
ownerSp (type: string)	Yes	The service provider owner.
firstUseDate (type: string)	Yes	Date when scratchcards was used. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
paymentModel (type: string)	No	Payment model
batchNumber (type: string)	Yes	The assigned batch number.

BETA

Element/Field	Required?	Description
category (type: string)	Yes	The category of the scratch card. Categories: CAT – classification PROD – Production MKTG – Marketing TEST - test
pinNumber (type: string)	Yes	Prepaid account code for scratch card
value (type: string)	Yes	The initial denomination of the scratchcards account.
loadingDate (type: string)	Yes	Loading date for scratch card. When the service delivery loads them into the database
status (type: string)	Yes	

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getScratchCards

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error gathering Scratchcards	If this error is returned, there was a problem with the Scratchsards IWS call. Please contact Iridium Service Delivery so they can investigate
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.69 getSecondaryDataService

The getSecondaryDataService request is used to get a specific secondary data service details. Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.

The getSecondaryDataService request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
id (type: string)	Yes	The internal ID for secondary data service.
serviceName (type: string)	No	The service name of a secondary data service.

A typical response will contain the following. (GetSecondaryDataServiceResponseImpl)

Element/Field	Required?	Description
secondaryDataService (type: secondaryDataServiceImpl)	No	This is the secondary data service available. This could be empty. See Secondary Data Service
permittedProviderAccounts	No	This is the list of providers that have access to the Secondary Data Service. This could be empty.
permittedProviderAccount (type: string)	No	Provider account number

13.70 getSecondaryDataServices

The getSecondaryDataServices request is used provide a list of secondaryDataServices based on search criteria. Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.

The getSecondaryDataServices request consists of the following pieces of data:

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
servicePlanId (type: string)	No	If desired to filter by service ID, specify the internal ID for service plan.

A typical response will contain the following. (GetSecondaryDataServicesResponseImpl)

Element/Field	**Required?**	**Description**
secondaryDataServicesMaps	Yes	List of secondary data service maps.
secondaryDataServicesMap (type: tns:secondaryDataServicesMapImpl)	No	This is the map of a secondary data service associated to specific service plan. This is not currently used. Secondary Data Services will be associated to all plans for the time being. This may be left be empty. See Secondary Data Services Map
independentSecondaryDataServices	Yes	List of secondary data services that are associated to ALL service plans for requesting provider.
secondaryDataService (type: tns: secondaryDataServiceImpl)	No	This is the list of providers that have access to the Secondary Data Service. This may be left empty be empty. See Secondary Data Service

BETA

13.71 getServicePlan

The getServicePlan request is used to get a specific service plan details. Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.

The getServicePlan request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
servicePlanId (type: string)	Yes	The internal ID for service plan.

A typical response will contain the following. (getServicePlanResponseImpl)

Element/Field	Required?	Description
servicePlan (type: servicePlanImpl)	No	This is the service plan available. This could be empty. See Service Plan

13.72 getServicePlans

The getServicePlans request is used to get a list of valid service plans. Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.

The getServicePlans request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
submarket (type: string)	No	SubMarket used to create or update contract. This will determine what plans are available based on SubMarket. If not passed it will pass back all available plans for the Partner.
model (type:string)	No	Used for Certus plans to get plan available by model.
isDemoTrial (type:datTypeEnum)	No	See Demo and Trial Types.
forActivate (type:Boolean)	No	Flag indicating whether the list returned is for a new activation or not. false – Return plans for update actions. true – Return plans eligible for new activations.

BETA

Element/Field	Required?	Description
fromPlanId (type:string)	No	
isPrepaidActive(type:Boolean)	No	Flag indicating if this request is intended for active prepaid (result list may differ depending on status of prepaid account) Default value is false true: Prepaid account exists false: Prepaid account does not exist
servicePlanType(type: servicePlanTypeEnum)	No	See Service Plan Type

A typical response will contain the following. (getServicePlansResponseImpl)

Element/Field	Required?	Description
submarkets	Yes	This is the list of SubMarkets available. This could be empty.
submarket (type: submarketImpl)	No	See Submarket

13.73 getSPNetGroups

The getSPNetGroups request is used to get a list of valid access groups.

The getSPNetGroups request consists of the following pieces of data: (getSPNetGroupsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
groupId (type: string)	No	Id of access group if desired to filter by id
groupName(type: string)	No	Best match for name of access group if desired to filter by name
globalOnly (type:string)	No	Include only public groups
status (type: spNetGroupStatusEnum)	No	Include groups based on current status

A typical response will contain the following. (getSPNetGroupsResponseImpl)

BETA

Element/Field	Required?	Description
Groups	Yes	List of access groups
group (type: groupImpl)	No	
id (type: string)	Yes	If of access group
name (type: string)	Yes	Name of access group
owner (type: string)	Yes	Owner of group
pages (type: pageImpl)	No	Specific page reference. See Pages
status (type: string)	Yes	Status of access group

BETA

13.74 getSPNetPages

The getSPNetPages request is used to get a list of SPNet pages.

The getSPNetPages request consists of the following pieces of data: (getSPNetPagesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
includeDetails (type: boolean)	Yes	TRUE: include page details FALSE: hide page details in response

A typical response will contain the following. (getSPNetPagesResponseImpl)

Element/Field	Required?	Description
Pages	Yes	This is the list of Pages available. This could be empty.
page (type: pageImpl)	No	See Pages

13.75 getSubmarkets

The getSubmarkets call is used to determine from SIM type and payment model. This is also used later to validateDeviceString for MSISDNs and possible SIMs. This currently will not need to be called for OpenPort or SBD services.

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
params (type: submarketSearchParametersImpl)	Yes	

Element/Field	Required?	Description
serviceType (type:serviceTypeEnum)	Yes	This is the service type enumeration for the correct service type. See Service Types.

A typical response will contain the following. (getSubMarketsResponseImpl)

Element/Field	Required?	Description
subMarkets	Yes	This is the list of SubMarkets available. This could be empty.
subMarket (type: subMarketImpl)	No	See Submarket

13.76 getSubscriberAccount

The Get Subscriber Account service allows a user to retrieve all of the details about a particular subscriber account. This includes the plan information, delivery destinations, MT filters, etc.

The getSubscriberAccount request consists of the following pieces of data. (getSubscriberAccountRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
accountNo (type:string)	Yes	The subscriber account number representing the account to search for.

A successful search will return a fully-populated subscriber account object that matches the account number the user searched for. The response will contain the following: (getSubscriberAccountResponseImpl)

Element/Field	Required?	Description
Account Object	Yes	One of the following account objects will be returned for a successful request. See Account for details regarding the details for each account type.
openPortSubscriberAccount (type: openPortSubscriberAccountImpl)	No	Provided if account type was Open Port
telephonySubscriberAccount (type: telephonySubscriberAccountImpl)	No	Provided if account type was Telephony

BETA

Element/Field	Required?	Description
pagingSubscriberAccount (type: pagingSubscriberAccountImpl)	No	Provided if account type was Paging
atsSubscriberAccount (type: atsSubscriberAccountImpl)	No	Provided if account type was ATS
burstDeviceSubscriberAccount (type: burstDeviceSubscriberAccountImpl)	No	Provided if account type was Burst Device
burstServiceSubscriberAccount (type: burstServiceSubscriberAccountImpl)	No	Provided if account type was Burst Service
sbdSubscriberAccount (type: sbdSubscriberAccountImpl)	No	Provided if account type was SBD
m2mSubscriberAccount (type: m2MSubscriberAccountImpl)	No	Provided if account type was M2M
pttDeviceSubscriberAccount (type: pttDeviceSubscriberAccountImpl)	No	Provided if account type was PTT Device
pttTalkGroupSubscriberAccount (type: pttTalkGroupSubscriberAccountImpl)	No	Provided if account type was PTT Talk Group
certusSubscriberAccount (type: certusSubscriberAccountImpl)	No	Provided if account type was Certus
Account Object	No	One type of Account object to be returned for this request. See Account for details regarding the details for each account type.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getSubscriberAccount

BETA

Errors or exceptions

Error	What it means/How to fix it
NOTE: The same error conditions apply on get subscriber account as they do on account searches. Please see those lists of errors for details. The following list will note error conditions that are different for this service.	
Either did not find a unique account matching search type [<search type>] and value [<value to search for>] or SP [XXXXXX] does not have access.	This error is returned for one of 3 reasons: - The account doesn’t exist. - The search criteria don’t match up (i.e., the search type is SUBSCRIBER_ACCOUNT_NUMBER, but a MSISDN was passed). - The SP specified by the serviceProviderAccountNumber does not have access to the subscriber being queried.

BETA

13.77 getSubscriberOrganizations

The Get Subscriber Org method will enable to the user to gather the array of subscriber Organizations linked to the provider account.

The getSubscriberOrgs request consists of the following pieces of data. (getSubscriberOrgsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Id (type:string)	No	Optionally enter the ID for the specific record is desired. If left blank an array of subscriber organizations will be returned.

A successful search will return an array of subscriber organization objects. The response will contain the following: (getSubscriberOrgsResponseImpl)

Element/Field	Required?	Description
Subscriber Organizations	No	An array of subscriber organizations
subscriberOrganization (type:subscriberOrganizationImpl)	No	The subscriber organization details See Subscriber Organization

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getSubscriberOrgs

13.78 getSubscriberOrgDetail

The Get Subscriber Org Detail method will enable to the user to retrieve the complete attributes associated with a subscriber organization. The getSubscriberOrgDetail request consists of the following pieces of data. (subscriberOrgDetailReqImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
subscriberOrgId (type:string)	Yes	The ID for the subscriber organization to search for.
deviceType (type: pttTypeEnum)	No	Optional parameter to limit the results to device objects or talkgroup objects. Default response will include array of both object types. See pttTypeEnum

A successful search will return the attributes of the subscriber organizations, as well as an array of device account details, and an array of talkgroup account details that are linked to the subscriber organization in the search criteria. The response will contain the following: (subscriberOrgDetailRespImpl)

Element/Field	Required?	Description
subscriberOrganization (type:subscriberOrgnaizationImpl)	Yes	
pttDevices	No	An array of PTT device account details.
device (type: pttDeviceDetailImpl)	No	
contractId (type:string)	No	
Imei (type:string)	No	
description (type:string)	No	
status (type:accountStatusEnum)	No	
activationDate (type:string)	No	
pttTalkgroups	No	An array of PTT talkgroup account details.

BETA

Element/Field	Required?	Description
talkgroup (type:pttTalkgroupDetailImpl)	No	
contractId (type:string)	No	
controllerId (type:string)	No	
status (type: accountStatusEnum)	No	
activationDate (type:string)	No	

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getSubscriberAccount

Errors or exceptions

Error	What it means/How to fix it
NOTE: The same error conditions apply on get subscriber account as they do on account searches. Please see those lists of errors for details. The following list will note error conditions that are different for this service.	

13.79 getSystemStatus

The getSystemStatus request is used to get the current status of the IWS system. This call can be useful to determine if the system is currently accepting requests or not.

The getSystemStatus request consists of the following pieces of data: (getSystemStatusResponseImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See [Authentication](#).

A typical response will contain the following. (getSystemStatusResponseImpl)

Element/Field	Required?	Description
status (type: systemStatusImpl)	Yes	The results of the status request
cbrmTime (type:string)	Yes	Current system time. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
openForBusiness (type:boolean)	Yes	TRUE – IWS accepting requests

BETA

Element/Field	Required?	Description
		FALSE – IWS currently not accepting requests.
maintenanceOccurring (type: boolean)	Yes	TRUE – IWS accepting requests FALSE – IWS currently not accepting requests.
outageReason (type:string)	Yes	**Description** for outage.
iwsVersion (type:string)	Yes	Current version running for IWS.
wsdlRevision (type:string)	Yes	Current WSDL version.
obsolete (type:boolean)	Yes	TRUE – WSDL is obsolete FALSE – WSDL is current
affectedElements	No	List of components currently not in an enabled status
affectedElement (type: string)	No	Affected Element
outageScheduleItems	No	List of future scheduled outages
outageScheduleItem(type: outageScheduleItemImpl)	No	Outage Item detail object
outageId (type:string)	Yes	Unique ID for the outage
startTime (type:string)	Yes	Start Time for outage
endTime (type:string)	Yes	End Time For Outage
name (type:string)	Yes	Name of scheduled outage.
description (type:string)	Yes	Description of scheduled outage item.
emergency (type:boolean)	Yes	TRUE – Emergency outage. FALSE – Scheduled outage.
Status (type: outageStatusEnum)	Yes	Current status of the outage
affectedElements	No	List of affected elements.
affectedElement (type:string)	No	Affected element name.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getSystemStatus

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.80 getVCSBundles

The getVCSBundles request is used to get a list of valid values for the VCS bundle field in the plan object. Note: These IDs may change from time to time and are not static. We advise against hard-coding these values. The VCS feature has not yet been enabled.

The getVCSBundles request consists of the following pieces of data: (getVCSBundlesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Valid plan types for this request include: openPortPlan:openPortPlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the VCS bundle that is already on the account. The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following. (getBundlesResponseImpl)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty.

BETA

Element/Field	Required?	Description
		bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
bundle (type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getVCSBundles

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.81 getVerticals

The getVerticals request is used to get a list of valid verticals.

The getVerticals request consists of the following pieces of data: (getVerticalsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.

A typical response will contain the following. (getVerticalsResponseImpl)

Element/Field	Required?	Description
Verticals	Yes	Array of Verticals
vertical (type: verticalImpl)	No	Vertical object
id (type: string)	No	Vertical ID
name (type: string)	No	Display Name
businessType (type: businessTypeEnum)	No	Business Type for Vertical

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getVerticals

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

1.1. getVessels

The get vessels call returns a list of vessels used for maritime provisioning.

The getVessels request consists of the following pieces of data: (getVesselsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
vessel (type: vesselImpl)	No	See Vessel
contractType (type: tns:contractTypeEnum)	No	See contractTypeEnum

A typical response will contain the following. (getVesselResponseImpl)

Element/Field	Required?	Description
vessel (type: vesselImpl)	Yes	See Vessel

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getVessels

13.82 getVoiceBundles

The getVoiceBundles request is used to get a list of valid values for the voiceBundleId field in the Plan object. *Note: These IDs may change from time to time and are not static. We advise against hard-coding these values.*

The getVoiceBundles request consists of the following pieces of data: (getVoiceBundlesRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
Plan	Yes	See Plan. Specify one Plan object, type depends on product desired. openPortPlan:openPortPlanImpl
fromBundleId (type:string)	Yes	When this request is for a new activation, this should be set to 0. Otherwise, it should be set to the value of the Voice bundle that is already on the account.

BETA

Element/Field	Required?	Description
		The reason for this is because the list of selections may be limited based on valid transitions.
forActivate (type:boolean)	Yes	Flag indicating whether the list returned is for a new activation or not. false - This is an existing subscriber and will be updated. true – This is a new activation.

A typical response will contain the following. (getBundlesResponse)

Element/Field	Required?	Description
Bundles	Yes	This is the list of bundles available based on the search criteria. This could be empty. bulkBundle:bulkBundleInfoImpl bundle:bundleInfoImpl datBundle:datBundleInfoImpl accountPoolBundle:accountPoolingBundleInfoImpl See Bundles.
bundle (type:bundleInfoImpl)	No	See Bundle.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getVoiceBundles

Errors or exceptions

Error	What it means/How to fix it
No Plan provided.	The plan object passed in on the request was null/empty. This is a required field and should contain all of your desired Plan data.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.83 getVoucherDenominations

This request will return the available voucher denominations for a given plan.

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
servicePlanId (type: string)	Yes	This is the servicePlan internal Id that identifies with plan the activation will use.
forActivate(type: boolean)	Yes	TRUE FALSE

A typical response will contain the following. (GetVoucherDenominationsResponseImpl)

Element/Field	Required?	Description
voucherDenominations	Yes	Array of voucher denomination. Can be empty.
voucherDenomination (type: voucherDenominationImpl)	No	Voucher Denomination Object.
displayName (type: string)	Yes	Display name for the voucher
voucherDenominationId (type: long)	Yes	Denomination ID associated with voucher denomination and type gathered.
voucherName (type: string)	Yes	Voucher name associated with voucher denomination
voucherDenomination (type: int)	Yes	Quantity of minutes or days for the voucher.
maxPurchase (type: int)	Yes	Max number of vouchers that can be purchased for that denomination.
extendActivation (type:string)	Yes	TRUE – time extensions allowed to be purchased with initial activation FALSE – extensions not permitted with activation purchase
extendRecharge (type:string)	Yes	TRUE – time extensions allowed to be purchased with recharge FALSE – extensions not permitted with recharge.
requiredTimeExtension (type: Boolean)	No	Indicates if time extensions are required on purchase of unit voucher.

BETA

Element/Field	Required?	Description
requiredTimeExtensionCount (type: int)	No	Indicates the number of required time extensions when requiredTimeExtension is true.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=getVoucherDenominations

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error gathering Voucher Denominations	If this error is returned, there was a problem with the Voucher Denominations IWS call. Please contact Iridium Service Delivery so they can investigate.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.84 loadIMEI

The loadIMEI method is used to load a new set of devices to be enabled for provisioning. This call is to be utilized by VAMs when producing new devices. Multiple devices can be loaded in a single request. All devices for a request must be the same model and manufacturer.

The loadIMEI consists of the following pieces of data: (loadIMEIRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
modelId (type: string)	Yes	Model ID to be associated to the devices
Devices		Array of devices to load
Device (type: string)	Yes	Device IMEI string (15 characters)
manufacturerId (type: string)	Yes	Manufacturer ID to be associated to the devices
Network (type: networkEnum)	Yes	Network Type for the devices

A typical response will contain the following. (loadIMEIResponseImpl)

Element/Field	Required?	Description
result (type: string)	Yes	Array of local number region choices
success (type: int)	No	Number of requested entries that succeeded.
failed (type: int)	No	Number of requested entries that failed.
details	Yes	Array of result details
detail (type: loadImeiErrorDetailImpl)	No	Detail entry of errors
device (type: string)	No	
error (type: string)	No	

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=loadIMEI

Errors or exceptions

Error	What it means/How to fix it
Unknown Error.	If this error is returned, there was some sort of a system failure or exception.

BETA

Error	What it means/How to fix it
	Please contact Iridium Service Delivery so they can investigate.

13.85 m2MImeiSearch

The m2mImeiSearch has been deprecated and will no longer be enabled. Please use createOnDemandReport or createSchedReport methods from the IWS Report endpoint in place of this request.

13.86 orderScratchcards

The orderScratchcards calls allows for a purchase of scratchcard orders in set quantities. The orderScratchcards request consists of the following pieces of data: (orderScratchcardsRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
dealId (type: string)	Yes	Scratchcard deal ID See getScratchcardDeals
quantity (type:string)	Yes	Quantity of cards

A typical response will contain the following. (orderScratchcardsResponseImpl)

Element/Field	**Required?**	**Description**
CertusScratchcardOrder (type: tns:certusScratchcardOrderImpl)	Yes	See Certus Scratchcard Order

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=orderScratchcards

13.87 rechargePrepaidAccount

The rechargePrepaidAccount allows a service provider to add a voucher for recharging a prepaid account.

The rechargePrepaidAccount request consists of the following pieces of data: (rechargePrepaidAccountRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.

BETA

msisdn (type: string)	Yes	Represents the phone number for VOICE calls. MSISDN-A is only used for prepaid.
accountNo (type:string)	Yes	
prepaidVoucher (type: prepaidVoucherimpl)	Yes	See Prepaid Voucher

A typical response will contain the following. (rechargePrepaidAccountResponseImpl)

Element/Field	**Required?**	**Description**
result (type: boolean)	Yes	Result of refund prepaid voucher request submission. The actual processing result will not be returned. True – Success False - Failed

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=rechargePrepaidAccount

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error gathering Recharge Prepaid Account	If this error is returned, there was a problem with the Recharge Prepaid Account IWS call. Please contact Iridium Service Delivery so they can investigate.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.88 refundPrepaidVoucher

The refund prepaid voucher allows a service provider to refund a voucher for a prepaid account based on SP, MSISDN, ICCID, or SERIAL NUMBER.

BETA

The refundPrepaidVoucher request consists of the following pieces of data: (refundPrepaidVoucherRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
msisdn (type: string)	Yes	Represents the phone number for VOICE calls. MSISDN-A is only used for prepaid.
iccid (type: string)	Yes	This is the number representing a SIM card. Here are some examples: 8988169**703**000007348 8988169**777**000016038 The bold numbers in the middle of the SIM number represent the network that the device belongs to. This same series of numbers is present in the MSISDNs that are activated on the account. When not applicable, it will be populated with “NA”.
serialNumber (type: string)	Yes	Serial number of the voucher being returned

BETA

A typical response will contain the following. (refundPrepaidVoucherResponseImpl)

Element/Field	Required?	Description
result (type: boolean)	Yes	Result of refund prepaid voucher request submission. The actual processing result will not be returned. True – Success False - Failed

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=refundPrepaidVoucher

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [<serviceProviderAccountNumber>].	A service provider associated with the specified account number could not be located. Please check that the value passed in the serviceProviderAccountNumber field is correct.
Error gathering Refund Prepaid Voucher	If this error is returned, there was a problem with the Refund Prepaid Voucher IWS call. Please contact Iridium Service Delivery so they can investigate.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

1.1. removeDIDWWNumberInventory

The removeDIDWWNumberInventory request is removes a local number from the inventory of both Iridium billing system and DIDWW The request consists of the following pieces of data. (removeDIDWWNumberInventoryRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See DIDWW Authentication.
localNumbers (type: string)	No	An array of local number device IDs
localNumber (type: string)	No	Local number device number

BETA

If the request is successful IWS will return a response object with a list of successfully removed devices and an error report object for any devices that should fail. (removeDIDWWNumberInventoryResponseImpl)

Element/Field	Required?	Description
successfulDevices (type: string)	Yes	An array of successfully removed local numbers
devices (type: xs:string)	No	
details (type: removeDIDWWErrorDetailImpl)	No	
detail (type: removeDIDWWErrorDetailImpl)	No	An array of error details
device (type:string)	No	Device that failed to be removed from DIDWW vendor
error (type: string)	No	Error reason for failure

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=removeDIDWWNumberInventory

13.89 resetIWSUserPassword

The resetIWSUserPassword function allows a service provider to change the secret key for their IWS user account. The resetIWSUserPassword request consists of the following pieces of data: (resetIWSUserPasswordRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
iwsUserId (type: string)	Yes	IWS user ID for account requesting password reset

A typical response will contain the following. (resetIWSUserPasswordResponseImpl)

Element/Field	Required?	Description
iwsUserId (type: string)	Yes	IWS user ID for account requesting password reset
iwsUserName (type: string)	Yes	
newPassword (type: string)	Yes	System generated new password value.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=resetIWSUserPassword

BETA

Errors or exceptions

Error	What it means/How to fix it
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Tier 2 Support so they can investigate.

13.90 resetUserPassword

The resetUserPassword function allows a service provider to unlock or reset an SPNet user account by changing the user password. The flow for this call is as follows:

- first call to resetUserPassword, first request does not have a token value supplied
- Email confirming request is sent to user email account
- User clicks link in email request (or data from link is passed back to IWS for second request)
- The email link contains a timestamp value and a token value, both are required for completion of the password reset action
- Second call made to IWS resetUserPassword, this time call requires token timestamp and token value.
- Password reset, account unlocked, email with new password sent to user.

BETA

The resetUserPassword request consists of the following pieces of data: (resetUserPasswordRequestImpl)

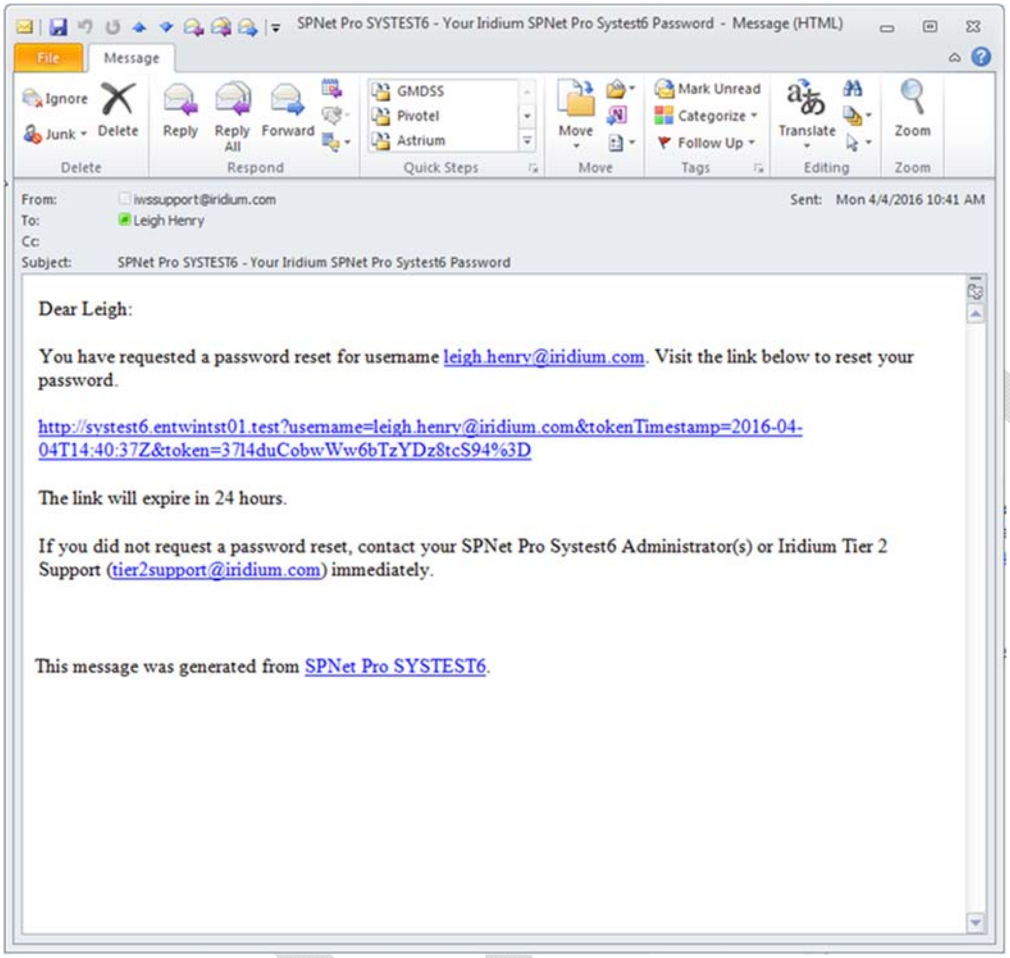
Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
userid (type: string)	No	SPNet user ID for account requesting password reset
username (type: string)	Yes	SPNet user name for account requesting password reset
tokenTimestamp (type: dateTime)	No	Timestamp value from original request. Leave blank during first request, required during second request.
Token (type:string)	No	Token value received from email after first request. Leave blank during first request, required during second request.

A typical response will contain the following. (resetUserPasswordResponseImpl)

Element/Field	**Required?**	**Description**
returnMessage (type: string)	Yes	Message indicating status of request

First request will return a message indicating email has been sent to user:

<returnMessage>An email has been sent to user *username* with further instructions for password reset.</returnMessage>



From: iwssupport@iridium.com

Sent: Mon 4/4/2016 10:41 AM

To: Leigh Henry

Cc:

Subject: SPNet Pro SYSTEST6 - Your Iridium SPNet Pro Systest6 Password

Dear Leigh:

You have requested a password reset for username leigh.henry@iridium.com. Visit the link below to reset your password.

http://systest6.entwintst01.test?username=leigh.henry@iridium.com&tokenTimestamp=2016-04-04T14:40:37Z&token=3714duCobwWw6bTzYDz8tcS94%3D

The link will expire in 24 hours.

If you did not request a password reset, contact your SPNet Pro Systest6 Administrator(s) or Iridium Tier 2 Support (tier2support@iridium.com) immediately.

This message was generated from SPNet Pro SYSTEST6.

Second response

<returnMessage>An email has been sent to user *username* with their new password.</returnMessage>

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=resetUserPassword

Errors or exceptions

Error	What it means/How to fix it
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Tier 2 Support so they can investigate.

13.91 resetTelephonyVMPin

The reset Telephony VM Pin function allows a service provider to modify the current VM Pin on a telephony account.

The resetTelephonyVMPin request consists of the following pieces of data: (resetTelephonyVMPinRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
accountNumber (type: string)	Yes	Account Number associated with the MSISDN to be modified.
msisdn (type: string)	Yes	Represents the phone number for VOICE calls.
newValue (type: int)	No	New value for pin. 4 digit number. Default: 9999

An empty response will indicate success.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=resetTelephonyVMPin

Errors or exceptions

Error	What it means/How to fix it
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Tier 2 Support so they can investigate.

BETA

13.92 restrictedSimSwap

The restrictedSimSwap method is reserved for future functionality

The restrictedSimSwap request consists of the following pieces of data. (getSubscriberAccountRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
subscriberAccountNumber (type:string)	Yes	The subscriber account number representing the account to swap sim.
deviceSwap (type: restrictedDeviceSwapImpl)	Yes	
oldDeviceNumber (type: string)	Yes	SIM (ICCID) of old sim.
newDeviceNumber (type: string)	Yes	SIM (ICCID) of new sim.

A successful search will return a fully-populated subscriber account object that matches the account number the user searched for. The response will contain the following: (restrictedSimSwapResponseImpl)

Element/Field	Required?	Description
Account Object	No	One type of Account object to be returned for this request. See Account for details regarding the details for each account type.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=restrictedSimSwap

13.93 sbdDeletePendingDownloads

This service will delete MT messages that are still PENDING (i.e., new or queued). It is meant to be used in conjunction with searchDeviceUsage.

The sbdDeletePendingDownloads request consists of the following pieces of data. (sbdDeletePendingDownloadsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
searchType (type:sbdSearchTypeEnum)	Yes	Which method should be used to lookup the account/device. See SBD Search Types.

BETA

Element/Field	Required?	Description
value (type:string)	Yes	Based on the search type specified, this should be the value representing the account to search for.
messageIds	No	This is a list of pending downloads.
messageId (type:string)	No	This is the actual ID of the download to delete. This can be retrieved from the results of the SBD Device Usage Report request when PENDING is the usage type passed in.
queuedMessage (type: Boolean)	Yes	

The response will contain the following: (sbdDeletePendingDownloadsResponseImpl)

Element/Field	Required?	Description
results	No	This is the list of results based on the attempted deletions. It will contain 1 result for each of the messageIds specified in the request.
result(type:sbdDeleteDownloadResultEntryImpl)	Yes	
messageId (type:int)	Yes	This is the actual ID of the message that was to be deleted.
result (type: sbdDeleteDownloadResultTypeEnum)	Yes	What the status is of the delete operation: See SBD Delete Download Result Types.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=sbdDeletePendingDownloads

BETA

Errors or exceptions

Error	What it means/How to fix it
One of the search parameters is null: type [<search type>] value [<value to search for>].	The searchType and value are both required parameters for this service. If this error is returned, it means that one of these required values was not populated.
Unable to locate service provider account [XXXXXX].	This means that the service provider represented by the account number XXXXXX could not be located. Thus, access to the results cannot be verified.
Unable to locate subscriber account with IMEI [XXXXXXXXXXXXXXX]. OR Unable to locate subscriber account with <search type> [<value to search for>].	This means that the subscriber could not be located (either via the IMEI or by some other search type like subscriber account number). Please ensure that the values passed are accurate.
One or both of the accounts are null and thus cannot perform lineage check - parent [XXXXXX], child [SUB-XXXXXX] OR Service provider [XXXXXX] is not a parent of account [SUB-XXXXXX].	These errors are returned if there are problems verifying access to the data being queried. Please check to ensure that the service provider account number and subscriber data are accurate. If there are still issues, contact Iridium Service Delivery for investigation.
The list of message IDs to delete is null or empty. Nothing more to do.	The list of message IDs to delete is a required field. If none are provided, this error is thrown. Please verify that this list is included.
An exception occurred deleting the pending downloads.	This error is returned if there are issues communicating with the provisioning web services. It is usually indicative of a system error. Contact Iridium Service Delivery for investigation.

BETA

13.94 sbdDestinationSearchReplace

This service will perform a global search and replace of delivery destinations. It is important to note that it is limited to replacing a configurable number of entries at a time (currently 250). If there are more than 250 entries, it will need to be executed more than once to change all occurrences of a destination value. This will also generate an email to the SPNet Notification address with the results of each execution of the service. This update request does not apply to deactivated accounts.

The sbdDestinationSearchReplace request consists of the following pieces of data. (deliveryDestinationSearchAndReplaceRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
oldValue (type:string)	Yes	The value to search for.
newValue (type:string)	No	The value to update all of the previous entries with.
searchOnlyFlag (type:boolean)	Yes	This flag can be used to trigger a search only capability when set to true. It is recommended to perform this search, ask the user to verify the results, and then request the service again with the flag set to false. If not specified, this flag will default to false.
includeSuspended (type: Boolean)	Yes	This flag can be used to indicate if the update request should apply to suspended accounts or not. TRUE – Apply update to suspended accounts FALSE – Do not apply updates to suspended accounts. Default is False.
serviceType (type: serviceTypeEnum)	No	Filter update action by device primary service.

BETA

The response will contain the following: (deliveryDestinationSearchAndReplaceResponseImpl)

Element/Field	Required?	Description
results	Yes	This is the list of results based on the attempted deletions. It may be empty.
result(type:sbdDestinationSearchReplaceEntryImpl)	No	
imei (type:string)	Yes	See *Account* for details on what this represents.
status (type:accountStatusEnum)	Yes	See *Account* for details on what this represents.
deliveryMethod (type:deliveryMethodTypeEnum)	Yes	See Delivery Methods for details on what this represents.
newDeliveryMethod (type:deliveryMethodTypeEnum)	No	See Delivery Methods for details on what this represents.
oldValue (type:string)	Yes	The value that was replaced during this operation.
newValue (type:string)	Yes	The value that replaced oldValue during this operation.
result (type:sbdDestSearchAndReplaceResultTypeEnum)	Yes	What the status is of the delete operation: See SBD Search and Replace Result Types.
configuredResultLimit (type:int)	Yes	This is the currently configured limit on the number of results to return for this search.
totalResultCount (type:int)	Yes	This is the total number of results that matched your search criteria. Could be much larger than the configuredResultLimit depending on wildcard use.

BETA

To generate sample requests and responses: https://training.iridium.com/iwsexplorer/index.php?call=sbdDestinationSearchReplace

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [XXXXXX].	This means that the service provider represented by the account number XXXXXX could not be located. Thus, access to the results cannot be verified.
The value to search for is null or empty. Please specify a valid search parameter.	The old value is a required parameter for this service. If this error is returned, it means that the old value was not populated.
There is nothing to do. The old value and new value are the same [<old value>].	This error is thrown if the oldValue and newValue passed into the request are the same value. Thus, there is nothing to do in this case.
The search value [<old value>] is not valid for any of the delivery method types.	This error is thrown if the oldValue does not match one of the known delivery method types. Please see Delivery Destination for a **Description** of the valid addresses.
An error occurred locating delivery destinations with value of [<old value>].	This error is returned if there are issues communicating with the billing system. It is usually indicative of a system error. Contact Iridium Service Delivery for investigation.
An exception occurred locating subscriber. OR An exception occurred deleting destination. OR An exception occurred adding filter.	This error is returned if there are issues communicating with the provisioning web services. It is usually indicative of a system error. Contact Iridium Service Delivery for investigation.

BETA

13.95 sbdMOQueuePurge

This function allows the user to purge pending downloads from the network by destination IP address. All pending messages to the specified server address, for devices belonging to the same provider account, will be purged.

The sbdMOQueuePurge request consists of the following pieces of data. (sbdMOQueuePurgeRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
deliveryMethodType (type: deliveryMethodTypeEnum)	Yes	See Delivery Method Type
address (type: string)	Yes	
network (type: homeGatewayEnum)	Yes	Select the Gateway to gather report details.

The response will contain the following: (sbdMOQueuePurgeResponseImpl)

Element/Field	**Required?**	**Description**
numberOfUpdatedRows (type: int)	Yes	The total number of rows updated.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=sbdMOQueuePurge

13.96 sbdMOQueueReport

This report will identify downloads from the network by destination address.

The sbdMOQueueReport request consists of the following pieces of data. (sbdMOQueueReportRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
deliveryMethodType (type: deliveryMethodTypeEnum)	Yes	See Delivery Method Type.
address (type: string)	Yes	
deliveryStatusType (type: deliveryStatusTypeEnum)	Yes	See Delivery Status Type.
network (type: homeGatewayEnum)	Yes	Select the Gateway to gather report details.

The response will contain the following: (sbdMOQueueReportResponseImpl)

BETA

Element/Field	Required?	Description
totalNumberOfRecords (type: int)	No	The total number of records returned by the search parameters.
Resources	Yes	This is the list of resources that were found based on the search criteria provided. It can be empty.
resource (type: moipQueueReportEntryImpl	Yes	
messageCount (type: int)	Yes	
deliveryMethodType (type: deliveryMethodTypeEnum)	Yes	See Delivery Method Type.
deliveryTo (type: string)	Yes	
deliveryStatusType (type: deliveryStatusTypeEnum)	Yes	See Delivery Status Type.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=sbdMOQueueReport

13.97 searchDeviceUsage

This report provides information on subscriber account regarding the device details, messages received and messages sent. A search can be performed on a subscriber account number or an IMEI number. The results for this search originate from DataMart.

The searchDeviceUsage request consists of the following pieces of data. (getDeviceUsageReportRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
scratchCardCriteria (type:scratchCardSearchCriteriaImpl)	No	Use this or SBD Search Criteria below
fromDate (type:string)	No	This is the date from which usage should be displayed. If none is specified, will default to January 1, 1970 00:00:00.000. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'

BETA

Element/Field	Required?	Description
pullActivity (type:boolean)	Yes	This flag will determine whether the activity details are pulled back in addition to the account summary. This will default to false. TRUE – pull activity FALSE – do not pull activity
toDate (type:string)	No	This is the date till which the usage should be displayed. If none is specified, will default to December 31, 9999 23:59:59.999. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
searchType (type: scratchCardSearchTypeEnum)	No	Which method should be used to lookup the account/device. See Scratch Card Search Types.
value (type:string)	No	Based on the search type specified, this should be the value representing the device being searched for.
sbdCriteria (type:sbdSearchCriteriaImpl)	No	Use this or Scratch Card Search Criteria above.
fromDate (type:string)	No	This is the date from which usage should be displayed. If none is specified, will default to January 1, 1970 00:00:00.000. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
pullActivity (type:boolean)	Yes	TRUE – pull activity FALSE – do not pull activity
toDate (type:string)	No	This is the date till which the usage should be displayed. If none is specified, will default to December 31, 9999 23:59:59.999. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
limit (type: int)	No	This will limit the number of results returned. If not specified, will default to the configured system limit (currently 1000).

BETA

Element/Field	Required?	Description
orderBy (type: sbdActivityOrderByTypeEnum)	No	This is an optional search parameter that can be used to sort the data by different values. If not specified, will default to UPLOAD_TIMESTAMP. See SBD Activity Order By Type. See the response below for information on what these fields are.
searchType (type:sbdSearchTypeEnum)	No	Which method should be used to lookup the account/device. See SBD Search Types.
startRow (type: int)	No	If paging is desired, this can be used in conjunction with the limit in order to pull result sets back in chunks.
usageSearchType (type:sbdUsageSearchTypeImpl)	No	This is an optional search parameter that can be used to search for different types of SBD usage. If not specified, will default to ALL. See SBD Usage Search Types.
value (type:string)	No	Based on the search type specified, this should be the value representing the device being searched for.

The response will contain the following: (getDeviceUsageReportResponseImpl)

Element/Field	Required?	Description
scratchCardDetails (type:scratchCardDetailsImpl)	Yes	Either scratchcard or SBD details will be returned depending on request type.
spAccountNumber (type:string)	Yes	Service Provider account number.
spName (type:string)	Yes	Service Provider Name.
accountNumber (type:string)	Yes	Internal account number for the scratch card.
accountStatus (type:accountStatusEnum)	Yes	Status of the internal scratch card account. See Account Status Types.

BETA

Element/Field	Required?	Description
type (type:string)	Yes	Recharge type of the voucher.
purchaseOrder (type:string)	No	Purchase order associated with this voucher.
serialNumber (type:string)	Yes	Serial number of this scratch card.
pin (type:string)	Yes	PIN number of this scratch card.
initialBalance (type:int)	Yes	Initial balance when the scratch card was purchased.
currentBalance (type:int)	Yes	Current balance of the voucher.
activationDate (type:string)	Yes	Date that the scratch card was activated. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
firstUseDate (type: string)	No	Date that the voucher was first used. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
expirationDate(type: string)	No	Date that the voucher expires. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
voucherStatus (type: string)	No	Shows the current status of the voucher.
activities	No	This is the list of activities that were found based on the search criteria provided. It can be empty if none were found or if pullActivity == false.
activity (type:voucherActivityImpl)	No	
imsi (type:string)	Yes	International Mobile Subscriber Identity See the following page for details on what an IMSI is: http://en.wikipedia.org/wiki/International_Mobile_Subscriber_Identity It is provided for information purposes only.

BETA

Element/Field	Required?	Description
imei (type:string)	Yes	International Mobile Equipment Identity This is the number that represents the physical hardware device. It is not required.
msisdn (type:string)	Yes	Number for originating call
callDate (type:string)	Yes	Timestamp of the voice or data call. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
calledNumber (type:string)	Yes	The number called.
callDuration (type:long)	Yes	Duration of the call.
ratedUnits (type:long)	Yes	Number of scratch card units consumed. This is generally 1 unit per second, but could be more or less depending on zoning or discounting.
countryName (type:string)	Yes	Country call was made in.
regionName (type:string)	Yes	Region Name for call origination
lac (type:string)	Yes	Location code for call origination
latMin (type:string)	Yes	Latitude Minutes
latPol (type:long)	Yes	Latitude Polarity
longMin (type:string)	Yes	Longitude Minutes
longPol (type:long)	Yes	Longitude Polarity
tZIndicator (type:int)	Yes	Time Zone indicator
tZOffset (type:string)	Yes	Time Zone offset
setAltitude (type:int)	Yes	Altitude of device at time of call
callingHC (type:int)	Yes	
callingVC (type:int)	Yes	
releaseCause (type: long)	Yes	Release Cause code for the call.

BETA

Element/Field	Required?	Description
sbdDetails (type:sbdDetailsImpl)	Yes	Either scratchcard or SBD details will be returned depending on request type.
totalNumberOfActivities (type:int)	No	This represents the total number of activity results that were found in the system (regardless of the limit imposed by the search).
activities	No	This is the list of activities that were found based on the search criteria provided. It can be empty.
activity (type:sbdActivityReportEntryImpl)	No	
mtActivity (type:sbdmtActivityImpl)	No	This object type contains details about a mobile terminated (MT) activity.
id (type:int)	Yes	This is a unique identifier for the activity entry.
imei (type:string)	Yes	This is the IMEI of the device.
size (type:string)	Yes	This is the size of the download.
deliveryStatus (type:string)	No	This is the status of the download.
timeReceived (type:string)	No	This is the time the device received the download. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
timeQueued (type:string)	No	This is the time the device queued the download. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
timeDelivered (type:string)	No	This is the time the device delivered the download. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
mtmsn (type:string)	No	This is the mobile terminated message sequence number.
moActivity (type:sbdmoActivityImpl)	No	This object type contains details about a mobile originated (MO) activity.

BETA

Element/Field	Required?	Description
id (type:int)	Yes	This is a unique identifier for the activity entry.
imei (type:string)	Yes	This is the IMEI of the device.
size (type:string)	Yes	This is the size of the upload.
deliveryStatus (type:string)	No	This is the status of the upload.
timeSent (type:string)	No	This is the time the device sent the upload. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
latitude (type: float)	No	This is the latitude representing where the device is physically located.
longitude (type:float)	No	This is the longitude representing where the device is physically located.
cep (type:int)	No	This is the Circular Error Probability to show accuracy.
momsn (type:string)	No	This is the mobile originated message sequence number.
deliveryMethod (type:deliveryMethodTypeEnum)	No	This is the delivery method. See Delivery Destination for details.
callStatus (type:string)	No	This is the status of the call.
Destination (type: string)	No	Destination for the call.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=searchDeviceUsage

DRAFT

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [XXXXXX].	This means that the service provider represented by the account number XXXXXX could not be located. Thus, access to the results cannot be verified.
One or both of the accounts are null and thus cannot perform lineage check - parent [XXXXXX], child [SUB-XXXXXX] OR Service provider [XXXXXX] is not a parent of account [SUB-XXXXXX].	These errors are returned if there are problems verifying access to the data being queried. Please check to ensure that the service provider account number and subscriber data are accurate. If there are still issues, contact Iridium Service Delivery for investigation.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

13.98 searchSimLock

This searchSimLock request is used to find the current simlock status of a particular IMEI.

The searchSimLock request consists of the following pieces of data. (searchSimLockRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
imei (type:string)	Yes	The IMEI to be searched.

If the request is successful, a response will be returned with the following data. (searchSimLockResponseImpl)

Element/Field	Required?	Description
results	No	An array of results.
result (type: simLockResultImpl)	No	
accountNumber (type: string)	No	Service Provider account number.
simLockStatus (type: simLockStatusEnum)	No	The status of the simLock. See SIM Lock Status.

BETA

13.99 searchVoucherOrder

This call is used to find a particular scratch card or E-voucher order entered into the Iridium system.

The searchVoucherOrder request consists of the following pieces of data. (searchVoucherInventoryRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
inventoryType (type:voucherInventoryTypeEnum)	Yes	See Voucher Inventory Types.
searchType (type:voucherOrderSearchTypeEnum)	Yes	See Voucher Order Search Types.
searchCriteria (type:string)	Yes	String holding the value to search for. This is either a PO number or a scratch card serial number depending on the searchType.

If the request is successful, a response will be returned with the following data. (searchVoucherInventoryResponseImpl)

Element/Field	Required?	Description
Orders	Yes	This object contains the results of the search. It could be empty if no results are found.
order (type:voucherOrderImpl)	No	This object contains the details of a record returned from the search.
purchaseOrder (type:string)	Yes	PO number for the order
description (type:string)	Yes	A Description entered when the order was created.
createDate (type:string)	Yes	Date that the order was entered
lineItems	No	List of line item objects
lineItem (type: voucherLineItemImpl)	No	
lineItemId (type: string)	No	
Part (type:bundleInfoImpl) OR bulkPart (type:bulkBundleInfoImpl)	Yes	
quantity (type:int)	Yes	Number of cards

BETA

Element/Field	Required?	Description
requestedStartSerial (type:string)	Yes	Start serial number requested with the order. In some cases this may not match the actual serial numbers assigned.
assignedSerialRanges	No	Array holding the ranges of serial numbers assigned to this order.
assignedSerialRange (type:serialRangeImpl)	No	
rangeEnd (type:string)	No	End serial number of a contiguous range.
rangeStart (type:string)	No	Start serial number of a contiguous range.
purchaseEventIds (type: string)	No	Purchase event identifier
batches	No	List of activation batches
batch (type:scratchCardActivationStatusImpl)	No	See Scratch Card Activation Status
requestId (type:string)	Yes	Request identifier
purchaseOrder (type:string)	No	PO Number
status (type:activationStatusEnum)	Yes	See Activation Status.
type (type:string)	Yes	See Activation Type.
startSerial (type:string)	Yes	The first serial number in the batch.
endSerial (type:string)	Yes	The final serial number in the batch.
requestTimestamp (type: string)	No	The time that the activation request was made.
totalRequested (type:int)	No	The number of scratch cards that were requested to be activated.
createdBy (type:string)	Yes	Date that the order was entered

13.100 setSubscriberAccountStatus

The setSubscriberAccountStatus request consists of the following pieces of data. (accountStatusChangeRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
updateType (type:statusChangeTypeEnum)	Yes	See Status Change Types.
value (type:string)	Yes	This should be the identifying number associated with the update Type specified above.
newStatus (type:accountStatusEnum)	Yes	This is the status the device/account will be changed to. See Account Status Types. NOTE: While the enumerated type contains PENDING, that will not affect the status of the account. PENDING is a derived state that indicates that an account is currently undergoing provisioning changes. If an account is already in this state, no updates will be permitted.
reason (type:string)	No	This is a comment field. To retrieve a list of common/recommended reason codes, the Get Reason Codes request can be used.

BETA

If the request is successful, an empty response will be returned.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=setSubscriberAccountStatus

Errors or exceptions

Error	What it means/How to fix it
Either did not find a unique account matching search type [<updateType>] and value [<value>] or SP [XXXXXX] does not have access.	This error is returned for one of 3 reasons: - The account doesn't exist. - The search criteria don't match up (i.e., the updateType is

Error	What it means/How to fix it
	SUBSCRIBER_ACCOUNT_NUMBER, but a MSISDN was passed). - The SP specified by the serviceProviderAccountNumber does not have access to the subscriber being queried. Please verify that the search criteria are correct and that the *subscriberAccountNumber* is correct.
The new status was null. Please specify a valid status.	The *newStatus* value was not set. Please specified the desired ending status.
Cannot set account status to PENDING. This is a derived state.	The *newStatus* value was specified as PENDING. The status cannot be directly set to PENDING. Please specify either ACTIVE, DEACTIVE, SUSPENDED, or CLOSED..
No status changes allowed on a CLOSED account.	The specified account is currently CLOSED. No further updates are allowed to closed accounts.
No status changes are allowed while service orders are still pending.	A previous update to this account resulted in the generation of one or more provisioning service orders. These service orders have not yet completed. No additional status changes are allowed until the pending service orders complete. Please retry the request later.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

13.101 subscribeBurstService

The subscribeBurstServicerequest is used to connect a device to a Burst service.

The subscribeBurstService request consists of the following pieces of data: (subscribeBurstServiceRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
imei (type: string)	Yes	IMEI for the device
serviceName (type: string)	Yes	Name for the service to subscribe.
GDA (type: string)	No	Geographic Data Area
BCA (type: string)	No	Broadcast Coverage Area
otaFlag (type: boolean)	Yes	TRUE – Over The Air enabled FALSE – Over the air disabled
expirationDate (type: string)	Yes	Date the subscription to the service will expire.

A typical response will contain the following (subscribeBurstServiceResponseImpl)

Element/Field	Required?	Description
subscribeBurstDeviceResponse (type: burstSubscriptionResponseImpl)	Yes	
purchaseOrder (type: string)	Yes	Purchase Order number for the request
requestId	Yes	Request Id for activation.
serialNumbers	No	Array of serial numbers
item (type:string)	No	Item Name subscribed to

13.102 unsubscribeBurstService

The unsubscribeBurstServicerequest is used to dis-connect a device from a Burst service.

The unsubscribeBurstService request consists of the following pieces of data: (unsubscribeBurstServiceRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
imei (type: string)	Yes	
serviceName (type: string)	Yes	

BETA

A successful result will return an empty response.

13.103 updateEmailNotification

The updateEmailNotification request will be utilized to update the settings for email notifications to the SP account.

The updateEmailNotification request consists of the following pieces of data

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
notifications (type: actionNotificationImpl)	No	List of notification name and enabled options
action (type: actionNotificationImpl)	Yes	Array of actions
actionType (type: emailNotificationTypeEnum)	Yes	Action type id. See Notification Action Types
description (type: string)	Yes	Display name of action type
name (type: string)	Yes	Name of notification option
enabled (type: string)	Yes	Is the notification option enabled? TRUE – Enabled FALSE - Disabled

A successful result will return an empty response.

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=updateEmailNotification

BETA

13.104 updatePTTSubOrgStatus

The Update PTT Subscriber Org Status method will enable to the user to update the status of a PTT subscriber organization.

The updatePTTSubOrgStatus request consists of the following pieces of data. (updatePTTSubOrgStatusRequestImpl)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
id (type:string)	Yes	Subscriber Organization ID
Refid(type:string)	No	Reference ID from controller
submarketId (type:string)	No	Submarket ID for the subscriber Organization
organizationName (type:string)	Yes	Name for Subscriber Organization
Description (type:string)	No	Optional field for user desired Description
createDate (type:dateTime)	No	
newStatus (type:statusEnum)	Yes	
reason (type:string)	No	

A successful response will contain the following: (updatePTTSubOrgStatusResponseImpl)

Element/Field	**Required?**	**Description**
returnMessage (type:string)	No	
ErrorMessage (type:string)	No	

To generate sample requests and responses: https://training.iridium.com/iwsexplorer/index.php?call=updateSubscriberOrg

13.105 updateQueueEntry

The Update Queue Entry method will enable the user to modify the status of a pending queue entry.

The updateQueueEntry request consists of the following pieces of data. (**updateQueueEntryRequestImpl**)

Element/Field	**Required?**	**Description**
Authentication Data	Yes	See Authentication.
queueEntryId (type:string)	Yes	The unique ID for the queue entry to be modified

BETA

Element/Field	Required?	Description
Status (type: queueEntryStatusEnum	Yes	Status value desired for queue update.

If the request is successful, an empty response will be returned.
To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=updateQueueEntry

BETA

13.106 updateSecondaryDataService

The updateSecondaryDataService request is used to update an existing secondary data service objects used for Certus provisioning.

The updateSecondaryDataService request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
secondaryDataService (type: secondaryDataServiceImpl)	Yes	Secondary Data Service to be updated. See Secondary Data Service
permittedProviderAccounts	No	This is the list of providers that have access to the Secondary Data Service.
permittedProviderAccount (type: string)	No	Provider account number

A typical response will contain the following. (GetSecondaryDataServiceResponseImpl)

Element/Field	Required?	Description
secondaryDataService (type: secondaryDataServiceImpl)	No	See Secondary Data Service
permittedProviderAccounts	No	This is the list of providers that have access to the Secondary Data Service.
permittedProviderAccount (type: string)	No	Provider account number

BETA

13.107 updateSubscriberOrganization

The Create Subscriber Org method will enable to the user to create a new subscriber organization for use with the PTT Controller.

The updateSubscriberOrg request consists of the following pieces of data. (updateSubscriberOrgRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
subscriberOrganization (type:subscriberOrganizationImpl)	Yes	The subscriber organization details See Subscriber Organization

A successful search will return a fully-populated subscriber organization object with updated values. The response will contain the following: (subscriberOrgsResponseImpl)

Element/Field	Required?	Description
Subscriber Organizations	No	An array of subscriber organizations
subscriberOrganization (type:subscriberOrganizationImpl)	No	The subscriber organization details See Subscriber Organization

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=updateSubscriberOrg

13.108 updateUserProfile

The updateUserProfile request is used to update the attributes of a user profile for a user account with SPNetPro access.

The updateUserProfile request consists of the following pieces of data: (updateUserProfileRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
currentUserName (type: string)	Yes	Username for the user to be modified.
updatedUser (type:userImpl)	Yes	User Object. See User.
resetPasswd (type:boolean)	Yes	TRUE – Password for user account will be reset. FALSE – Password will not be modified.

A typical response will contain the following (updateUserProfileResponseImpl)

Element/Field	Required?	Description
user (type: userImpl)	Yes	The user object created. See User.

BETA

1.1. updateVessels

The update vessels call is used to update an existing vessel used for maritime provisioning.

The updateVessels request consists of the following pieces of data: (updateVesselsRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
vessel (type: vesselImpl)	No	See Vessel
obsolete (type:boolean)	No	Mark vessel of obsolete

A typical response will contain the following. (updateVesselResponseImpl)

Element/Field	Required?	Description
vessel (type: vesselImpl)	Yes	Return vessel with updated fields See Vessel

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=updateVessels

13.109 userAuthenticate

The userAuthenticateRequest request authenticate user credentials.

The userAuthenticateRequest request consists of the following pieces of data: (userAuthenticateRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
username (type: string)	Yes	Username for the user to be authenticated.
password (type:string)	Yes	Password for user account.

If the request is successful, the account status will be modified and an empty response will be returned.

BETA

13.110 unlockUser

The unlockUser request is used to reset the lock indicator for an SPNet user account. This call should be used by SP admin users that desire to unlock a user account when the password is known. If the password is unknown the user should initiate the resetPassword feature.

The unlockUser request consists of the following pieces of data: (userUnlockRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
user (type: string)	Yes	Username for the user to be modified.

If the request is successful, the account status will be modified and an empty response will be returned.

13.111 unquarantineDevice

The unquarantineDevice request consists of the following pieces of data. (unquarantineDeviceRequestImpl)

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
deviceNumber (type: string)	Yes	Numeric value of the device to be unquarantine

If the request is successful, an empty response will be returned.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=unquarantineDevice

13.112 validateDeviceString

The validateDeviceString request is used to get the validity of a device ID before activation.

The validateDeviceString request consists of the following pieces of data:

Element/Field	Required?	Description
Authentication Data	Yes	See Authentication.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
deviceString (type:string)	Yes	The device string value that will be validated by this call.
deviceStringType (type:deviceStringTypeEnum)	Yes	The type of device that will be checked against the string value of deviceString. See deviceStringTypeEnum.

BETA

Element/Field	Required?	Description
fromDeviceString (type:string)	No	For an update, this would be the device that the account currently has provisioned, i.e., the existing SIM number. If the SIM is being swapped, IWS will use this information when validating the new SIM to make sure that it is on the same network as the old one. Currently, for SBD, this does not impact the result of the request and can be left as an empty string.
validateState (type:boolean)	No	true false This Boolean value states whether or not to check if the device is in use by another contract or in state that is not usable. If false then it only checks valid format.
fromSubmarket (type: string)	No	Used to validate whether or not a device is in a specific submarket.

A typical response will contain the following. (validateDeviceStringResponseImpl)

Element/Field	Required?	Description
validSimDeviceData (type:validSimDeviceDataImpl)	No	Will be provided for SIM validation This could be empty.
valid (type:string)	Yes	True false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
deviceSim (type: deviceSimImpl)	No	This is the actual object that represents the SIM being validated. See SIM. NOTE: It will not be included for an invalid device string.
validImeiDeviceData (type: validImeiDeviceDataImpl)	No	Will be provided for IMEI validation This could be empty.

BETA

Element/Field	Required?	Description
valid (type:string)	Yes	true false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
imeiProfile (type: imeiProfileImpl)	No	This is the actual object that represents the IMEI being validated.
serviceType (type: serviceTypeEnum)	Yes	See Service Types.
tac (type: string)	No	
fac (type: string)	No	
maxMessageSize (type: int)	No	Maximum message size the device can support
description (type: string)	No	
flags (type: imeiTypeEnum)	No	
model (type: string)	No	Model ID for the device
imeiSimBasedFlag (type:string)	No	true – this model can support a sim false – this model cannot support a sim
serviceCategory (type: modelServiceCategoryEnum)	No	See Model Service Category
submarket (type: string)	No	Submarket associated to the device
vertical (type: verticalImpl)	No	Vertical linked to the device
id	No	Id for the vertical classification
name (type: string)	No	Name of the vertical
businessType (type: businessTypeEnum)	No	Business type for the vertical

BETA

Element/Field	Required?	Description
validMsisdnDeviceData (type:validMsisdnDeviceDataImpl)	No	Will be provided for MSISDN validation This could be empty.
valid (type:string)	Yes	true false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
deviceNumber (type: deviceNumberImpl)	No	This is the actual object that represents the MSISDN being validated. See Generic Number. NOTE: It will not be included for an invalid device string.
validMsisdnbDeviceData (type:validMsisdnDeviceDataImpl)	No	Will be provided for MSISDN-B validation This could be empty. This device type is available for support of Russian devices.
valid (type:string)	Yes	true false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
deviceNumber (type: deviceNumberImpl)	No	This is the actual object that represents the MSISDN being validated. See Generic Number. NOTE: It will not be included for an invalid device string..
validMsisdncDeviceData (type:validMsisdncDeviceDataImpl)	No	Will be provided for MSISDN-C validation

BETA

Element/Field	Required?	Description
		This could be empty.
valid (type:string)	Yes	true false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
deviceNumber (type: deviceNumberImpl)	No	This is the actual object that represents the MSISDN-C being validated. See Generic Number. NOTE: It will not be included for an invalid device string.
validIsdnaDeviceData (type:validIsdnaDeviceDataImpl)	No	Will be provided for ISDNA validation This could be empty.
valid (type:string)	Yes	true false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
deviceNumber (type: deviceNumberImpl)	No	This is the actual object that represents the ISDNA being validated. See Generic Number. NOTE: It will not be included for an invalid device string.
validCapcodeDeviceData (type:validCapcodeDeviceDataImpl)	No	Will be provided for Capcode validation This could be empty.
valid (type:string)	Yes	true false

BETA

Element/Field	Required?	Description
		This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
deviceCapcode (type: deviceCapcodeImpl)	No	See Capcode.
validIpDeviceData (type:validIpDeviceDataImpl)	No	Will be provided for IP Address validation This could be empty.
valid (type:string)	Yes	true false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
deviceIp (type: deviceIpImpl)	No	See IP Addr.
validIcaoData (type:validIcaoDataImpl)	No	Will be provided for ICAO validation This could be empty.
valid (type:string)	Yes	True false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
validTailNumData (type:validIpDeviceDataImpl)	No	Will be provided for Tail Number validation This could be empty.
valid (type:string)	Yes	True

BETA

Element/Field	Required?	Description
		false This indicates whether or not the device string is valid.
deviceString (type: String)	Yes	This is the string that was validated
reason (type: String)	No	The reason if the string is not valid.
homeGateway (type: homeGatewayEnum)	No	This is the default Home Gateway built into the device.
submarket (type: submarketImpl)	No	This is the specific submarket for telephony and paging. See Submarket

To generate sample requests and responses:
https://training.iridium.com/iwsexplorer/index.php?call=validateDeviceString

Errors or exceptions

Error	What it means/How to fix it
Unable to validate device string [<deviceString>]. The device string type is null.	The request did not specify a deviceStringType.
Unsupported device string type [<deviceStringType>].	An unrecognized value for the deviceStringType was specified. Valid values are SIM, and IMEI.
Unknown Error.	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

14 Reusable Objects

IWS was designed using an object-oriented approach for the interface. For this reason, there are common elements that are utilized across many of the requests and responses. The objects listed below will appear throughout the services and should become familiar:

14.1 Scratch Card Activation Status

The scratch card activation status object includes the following data:

Element/Field	Required?	Description
activationList	No	Array of activation requests

activationStatus (type:scratchCardActivationStatusImpl)	No	Activation object.
requestId (type:string)	Yes	Request identifier
purchaseOrder (type:string)	No	PO Number
Status (type:activationStatusEnum)	Yes	Status of the activation request. Possible values are: PENDING, FINISHED, or ERROR
Type (type: activationTypeEnum)	Yes	Type of activation. Possible values are: FULL, LINEITEM, or RANGE
startSerial (type:string)	Yes	The first serial number in the batch.
endSerial (type:string)	Yes	The final serial number in the batch.
requestTimestamp (type:string)	No	The time that the activation request was made. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
totalRequested (type:int)	No	The number of scratch cards that were requested to be activated.

To generate sample requests and responses use:
https://training.iridium.com/iwsexplorer/index.php?call=searchVoucherOrder

BETA

Errors or exceptions

Error	What it means/How to fix it
Unable to locate service provider account [XXXXX].	The Service Provider account number that was provided is incorrect and no account could be found.
No valid search criteria provided.	The search type must be SERIAL_NUMBER, PIN, or ACCOUNT_NUMBER.
Error retrieving activity details.	A system error occurred. Try your search again later. If this error persists, please contact Iridium Service Delivery so they can investigate.
Invalid date range: start [XXXX] end [YYYY].	The dates given are invalid. Double check and correct them.
Error retrieving device.	A system error reading the voucher information occurred. Try your search again later. If this error persists, please contact Iridium Service Delivery so they can investigate.
Service Provider is required to perform this search.	A Service Provider account must be provided.
Cannot find scratchcard device for serial XXXX	The provided serial number is invalid.
Unknown Error	If this error is returned, there was some sort of a system failure or exception. Please contact Iridium Service Delivery so they can investigate.

BETA

14.2 Account

The account object contains all of the data necessary to describe a provisioned subscriber. It contains the plan/pricing selections and the provisioning details (devices, phone line options, etc.).

14.2.1 Base Account Object

The following object definition represents the base Account object (absSubscriberAccountImpl).

Element/Field	Required?	Description
subscriberAccountNumber (type:string)	Yes (is nillable - see Description)	Contains the actual account number which represents the subscriber account. Follows the convention “SUB-XXXXXXX”. This field will be generated during Activations and returned on the

Element/Field	Required?	Description
		response. It will be populated on the response for Get Subscriber Account. It is required on the request for Updates to let IWS know which account we are modifying.
accountStatus (type:accountStatusEnum)	Yes (is nillable - see Description)	Represents the current status of the subscriber account. This field will be ignored on activation requests. It will be populated on the response for GetSubscriberAccount. It can be used to change the status of the account on Updates. NOTE: If modified as part of an update, other changes on the same request may be ignored. The preferred method for updating the status of an account is described in the section titled Deactivations and Suspensions. See Account Status.
statusDescription (type:string)	Yes (is nillable - see Description)	Description of the last status change. It is a system generated field that is a concatenation of the time it was changed and the reason given for the change. This field cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
createDate (type:string)	No	This is the date the account was created. It is a system generated field. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format.
updateDate (type:string)	No	This is the date the account was last modified. It is a system generated field. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.

BETA

Element/Field	Required?	Description
		Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format.
closedDate (type:string)	No	This is the date the account was closed/deactivated. It is a system generated field. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account if the account is in a DEACTIVE state. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format.
createdBy (type:string)	No	This is the name of the individual that created the account. It is set during activations based on the caller that is passed in the base request. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
updatedBy (type:string)	No	This is the name of the individual that last modified the account. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.

BETA

14.2.2 ATS Account

Element/Field	Required?	Description
atsSubscriberAccount (type: atsSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
icao (type: string)	Yes	This is the ICAO Description identifier for the ATS account.
regTailNumber (type: string)	Yes	This is the Description of the Registration Tail number found. This is a unique identifier that is unique and searchable.
airline (type: string)	Yes	This is enumerated type of names passed back from getAirlines call See getAirlines.

Element/Field	Required?	Description
activateSecondary (type: Boolean)	Yes	True – secondary account desired and included in activation request. False – secondary account not desired, account object will be ignored if supplied in activation request.
TelephonySubscriberAccountPrimary (type: telephonySubscriberAccountImpl)	Yes	See Telephony Account. Primary voice channel
TelephonySubscriberAccountSecondary (type: telephonySubscriberAccountImpl)	No	See Telephony Account. Secondary voice channel

14.2.3 Burst Device Account

The following object definition represents the Burst Device Account object (burstDeviceSubscriberAccountImpl).

Element/Field	Required?	Description
burstDeviceSubscriberAccount (type: burstDeviceSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
commitmentStart (type:string)	No	This is the date the commitment obligation started for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
commitmentEnd (type:string)	No	This is the date the commitment obligation ends for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format.

BETA

Element/Field	Required?	Description
		It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
plan (type: burstDevicePlanImpl)	Yes	See Burst Device Plan.
subscriberAccountMetadata (type: burstDeviceSubscriberAccountMetadataImpl)	No	Provider reference data for the account.
spReference (type:string)	No	This is a freeform field for providers to enter reference data in regards to the SIM, for example a PO number, or customer code. This field will be searchable in accountSearch method.
imei (type: string)	No	International Mobile Equipment Identity This is the number that represents the physical hardware device.
subscriptions	No	Array of subscriptions for the device
subscription (type: burstSubscriptionImpl)	No	Subscription object
serviceName(type: string)	No	
GDA(type: string)	No	Geographic Destination Area
BCA(type: string)	No	Broadcast Coverage Area
otaFlag(type: string)	No	TRUE – Over The Air enabled FALSE – Over the air disabled
expirationDate(type: string)	No	Date the subscription to the service will expire.
subscribedDate(type: string)	No	Date the subscription to the service was requested.
status (type: accountStatusEnum)	No	Status of subscription

BETA

14.2.4 Burst Service Account

The following object definition represents the Burst Service Account object (burstServiceSubscriberAccountImpl).

Element/Field	Required?	Description
burstServiceSubscriberAccount (type: burstServiceSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
serviceName (type: string)	No	Service Name
plan (type: burstServiceSubAccountPlanImpl)	Yes	See Burst Service Plan.
Subscribers	No	Array of subscribers to this service.
Subscriber (type:burstSubscriberImpl)	No	Individual Subscriber Details
Imei (type: string)	No	
GDA(type: string)	No	Geographic Destination Area
BCA(type: string)	No	Broadcast Coverage Area
otaFlag(type: string)	No	TRUE – Over The Air enabled FALSE – Over the air disabled
expirationDate(type: string)	No	Date the subscription to the service will expire.
subscribedDate(type: string)	No	Date the subscription to the service was requested.
status (type: accountStatusEnum)	No	Status of subscription
serviceType (type: serviceTypeEnum)_	No	Service type of the device subscribed

BETA

14.2.5 Certus Account

The following object definition represents the Certus Account object (certusSubscriberAccountImpl).

Element/Field	Required?	Description
certusSubscriberAccount(type: certusSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
commitmentStart (type:string)	No	This is the date the commitment obligation started for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
commitmentEnd (type:string)	No	This is the date the commitment obligation ends for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
plan (type: certusPlanImpl)	Yes	See Certus Plan.
sim (type:deviceSimImpl)	Yes	See SIM. While the entire object is included on the account, activations and updates only require the device number to be specified (however, the whole object is acceptable). The object will be populated on calls to Get Subscriber Account. There is a web service that allows AJAX validation of this field. See Validate Device String.

BETA

Element/Field	**Required?**	**Description**
Imei (type:certusImeiImpl)	Yes	Object containing information related to the IMEI
deviceNumber (type:string)	Yes	International Mobile Equipment Identity This is the number that represents the physical hardware device. It is required.
model (type:string)	No	Certus model associated to the IMEI
serviceCategory (type:string)	No	Service category as it exist on the contract model
serviceClass (type:string)	No	Service class as it exist on the contract model
maxDataSpeed (type:int)	No	Max data speed on contract. Can differ from model if the model has been updated by contract firmware as not.
homeGateway (type:homeGatewayEnum)	No	This value will be present for customers with the capability to override default home gateway on SIM-based IMEIs on SBD subscriber accounts. This is not available for non SIM-based IMEIs, or for customers without home gateway override privileges.
ip (type:deviceIpImpl)	Yes	See Ip. While the entire object is included on the account, activations and updates only require the device number to be specified (however, the whole object is acceptable). The object will be populated on calls to Get Subscriber Account. IWS will also accept an empty string as the device number, which will trigger the auto-assign capability.
publicIP (type:string)	No	This is the static public IP to be utilized if public ip network has been selected.
simImeiLock (type:string)	No	True – two way lock between sim andIMEIwill be enforced False – no lock betweenSIMand IMEI

BETA

Element/Field	Required?	Description
safetyEnabled (type:string)	No	True – safety services enabled (May involve additional charges) False – safety services not enabled
atsSafetyInfo (type: atsSafetyInfoImpl)	No	ATS safety information
icao (type:string)	No	ICAO for aircraft – Required for aviation contracts
regTailNumber (type:string)	No	Tail number / registration number of aircraft Required for aviation contracts
airline (type:string)	No	ID for airline of plane. Required if safety enabled.
maritimeSafetyInfo (type: maritimeSafetyInfoImpl)	No	See Maritime Safety Info
scratchCardsRestricted (type:boolean)	No	true – This will limit scratch card usage to cards that are owned by the same service provider as the OpenPort device itself. false - This will allow any Iridium OpenPort scratch card to be used on the account.
subscriberAccountMetadata (type: certusSubscriberAccountMetadataImpl)	No	Provider reference data for the account.
spReference (type:string)	No	This is a freeform field for providers to enter reference data in regards to the SIM, for example a PO number, or customer code. This field will be searchable in accountSearch method.
primaryDataServices	No	Array of primary data services
primaryDataService(type: primaryDataServiceImpl)	No	Details for Primary Data Service See Primary Data Service
optionalSecondaryDataServices	No	Array of optional secondary data services

BETA

Element/Field	Required?	Description
optionalSecondaryDataService(type: optionalSecondaryDataServiceImpl)	No	Details for Primary Data Service See Optional Secondary Data Service
voiceToggle (type: string)	No	True – enable voice services False – voice services disabled
voiceServiceDetail (type: certusVoiceImpl)	No	Voice service details
status (type: accountStatusEnum)	No	Status of voice service
bundleOptionIds	No	List of standalone bundles for Voice Service.
bundleOptionId (type: string)	No	Id of standalone bundle deal
voiceLines	No	Array of voice lines
voiceLine (type: certusDeviceNumberImpl)	No	Details for voice line See Certus Voice Lines
streamingToggle (type: string)	No	True – enable streaming services False – streaming services disabled
streamingServiceDetail (type: certusStreamingImpl)	No	Details for streaming options if enabled
status (type: accountStatusEnum)	No	Status of streaming service
bundleOptionIds	No	List of standalone bundles for Streaming Service.
bundleOptionId (type: string)	No	Id of standalone bundle deal
streamingSpeed (type: int)	No	Must be between 1 and the max aggregate rate on the model
sbdToggle (type: string)	No	True – enable SBD services False – SBD services disabled
sbdServiceDetail (type: certusSbdImpl)	No	Details for SBD options if enabled.
status (type: accountStatusEnum)	No	Status of SBD service

BETA

Element/Field	Required?	Description
bundleOptionIds	No	List of standalone bundles for SBD Service.
bundleOptionId (type: string)	No	Id of standalone bundle deal
lritFlagstate (type:Boolean)	No	True – enable LRIT services False – LRIT services disabled
deliveryDetails	Yes	List of Delivery Destinations. Must contain at least 1, but no more than 5 entries.
deliveryDetail (type: deliveryDestinationImpl)	Yes	List of Delivery Destinations. Must contain at least 1, but no more than 5 entries. See Delivery Destination.
mtFilters	Yes	List of MT Filters. May contain up to 10 entries.
mtFilter (type: mtFilterImpl)	Yes	See MT Filter.
burstToggle (type: string)	No	True – enable Burst services False – Burst services disabled
burstServiceDetail (type: certusBurstImpl)	No	Details for Burst options if enabled.
status (type: accountStatusEnum)	No	Status of Burst service
bundleOptionIds	No	List of standalone bundles for Burst Service.
bundleOptionId (type: string)	No	Id of standalone bundle deal
subscriptions	Yes	Array of subscriptions associated to the burst service
subscription (type: burstSubscriptionImpl)	Yes	
serviceName(type: string)	No	
GDA(type: string)	No	Geographic Destination Area
BCA(type: string)	No	Broadcast Coverage Area

BETA

Element/Field	Required?	Description
otaFlag(type: string)	No	TRUE – Over The Air enabled FALSE – Over the air disabled
expirationDate(type: string)	No	Date the subscription to the service will expire.
subscribedDate(type: string)	No	Date the subscription to the service was requested.

14.2.6 Certus Scratchcard

The following object definition represents the Certus Scratchcard object (CertusScratchcardImpl).

Element/Field	Required?	Description
CertusScratchcard(type: CertusScratchcardImpl)	No	
serialNumber (type:string)	Yes	15 digit identifier
pin (type:string)	Yes	Scratchcard PIN number used for making calls
orderNo (type:long)	Yes	Order ID scratchcard belongs to
dataPlanId (type:int)	Yes	Deal ID for scratchcard
dataPlanName (type:string)	Yes	Name of the deal Id
status (type: ScratchcardStatusEnum)	Yes	Status of the scratchcard. See scratchcardStatusEnum
activationDate (type: String)	Yes	Date of activation in billing system

BETA

14.2.7 Certus Scratchcard Order

The following object definition represents the Certus Scratchcard Order object (CertusScratchcardOrderImpl).

Element/Field	Required?	Description
CertusScratchcardOrder(type: CertusScratchcardOrderImpl)	No	
orderId (type: String)	Yes	Primary ID of an order
dealId (type:long)	Yes	Deal associate to the order
quantity (type: int)	Yes	Quantity of cards in the order
startSerialNumber (type: String)	Yes	Starting serial number
endSerialNumber (type: String)	Yes	Ending serial number
createdBy (type: String	No	
createdDate	No	
Status (type: ScratchcardOrderStatusEnum)	Yes	See scratchcardStatusEnum
certusScratchcards	No	An array of scratchcards that have been request for activation.
Scratchcard (type: CertusScratchcardImpl)	No	See Certus Scratchcard

BETA

14.2.8 Primary Data Service

The following object definition represents the Certus Primary Data Service object (PrimaryDataServiceImpl).

Element/Field	Required?	Description
PrimaryDataService (type: PrimaryDataServiceImpl)	No	
serviceName (type: primaryDataServiceTypeEnum)	Yes	Name of primary data service. See Primary Data Service Type
serviceEnabled (type:boolean)	Yes	Turn the service on or off.
queueWeight	No	This is a placeholder value for future release. Currently any value submitted will be ignored.

Element/Field	Required?	Description
pepEnabled(type: secondaryDataServiceOverrideEnum)	No	0 = Default for service; 1 = Enabled; 2 = Disabled See Secondary Data Service Override
hdrCompEnabled(type: secondaryDataServiceOverrideEnum)	No	0 = Default for service; 1 = Enabled; 2 = Disabled See Secondary Data Service Override
payloadCompEnabled(type: secondaryDataServiceOverrideEnum)	No	0 = Default for service; 1 = Enabled; 2 = Disabled See Secondary Data Service Override

14.2.9 Optional Secondary Data Service

The following object definition represents the Certus Primary Data Service object (optionalSecondaryDataServiceImpl).

Element/Field	Required?	Description
optionalSecondaryDataService(type: optionalSecondaryDataServiceImpl)	No	
secondaryDataServiceId(type: string)	Yes	ID value of secondary data service.
serviceName(type: string)	No	Identifies the data service for which the flow is to be used
idcpId(type: string)	Yes	Valid values 0-6. Identifies IDCP context over which the IP flow is carried
pepEnabled(type: secondaryDataServiceOverrideEnum)	No	0 = Default for service; 1 = Enabled; 2 = Disabled See Secondary Data Service Override
hdrCompEnabled(type: secondaryDataServiceOverrideEnum)	No	0 = Default for service; 1 = Enabled; 2 = Disabled See Secondary Data Service Override
payloadCompEnabled(type: secondaryDataServiceOverrideEnum)	No	0 = Default for service; 1 = Enabled; 2 = Disabled See Secondary Data Service Override

BETA

Element/Field	Required?	Description
billingIdOverride(type: int)	No	This is a placeholder value for future release. Currently any value submitted will be ignored.
expeditedRateCap(type: int)	No	Only used if HrlpSapi=0 Sets the maximum number of bits per second for which the flow has EXPEDITED priority. Any excess demand will be handled only after all NORMAL priority flows This is a placeholder value for future release. Currently any value submitted will be ignored.
queueWeight(type: int)	No	Only Used if HrlpdSapi=1 Sets the weigthing value used when multiple NORMAL priority flows are contending for bandwidth. (Valid Range 1-100) This is a placeholder value for future release. Currently any value submitted will be ignored.

BETA

14.2.10 Maritime Safety Info

The following object definition represents the Maritime Safety Info object. (maritimeSafetyInfoImpl).

Element/Field	Required?	Description
maritimeSafetyInfo (type: maritimeSafetyInfoImpl)	No	
Vessel (type:vesselImpl)	Yes	See Vessel
safetyType (type: vesselLinkTypeEnum)	Yes	See vesselLinkTypeEnum

14.2.11 M2M+ Account

The following object definition represents the M2M+ Account object (m2mSubscriberAccountImpl).

Element/Field	Required?	Description
m2mSubscriberAccount (type:m2mSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
plan (type: m2mPlanImpl)	Yes	See M2M Plan.
subscriberAccountMetadata (type: m2mSubscriberAccountMetadataImpl)	No	Provider reference data for the account.
spReference (type:string)	No	This is a freeform field for providers to enter reference data in regards to the SIM, for example a PO number, or customer code. This field will be searchable in accountSearch method.
imei (type:string)	Yes	Must be a valid IMEI for an SBD device. At a very basic level, it should be 15-digit number starting with a 3 and ending with a 0. There is a web service that allows AJAX validation of this field. See Validate Device String.
homeGateway (type:homeGatewayEnum)	No	This value will be present for customers with the capability to override default home gateway onSIMbased IMEIs on SBD subscriber accounts. This is not available for non SIM-based IMEIs, or for customers without home gateway override privileges.
msisdnc (type:deviceImpl)	Yes	See Telephony Device. While the entire object is included on the account, activations and updates only require the device number to be specified (however, the whole object is acceptable).

BETA

Element/Field	Required?	Description
		The object will be populated on calls to Get Subscriber Account. IWS will accept an empty string as the device number, which will trigger the auto-assign capability.
msisdnb (type:deviceImpl)	Yes	See Telephony Device. While the entire object is included on the account, activations and updates only require the device number to be specified (however, the whole object is acceptable). The object will be populated on calls to Get Subscriber Account. IWS will accept an empty string as the device number, which will trigger the auto-assign capability.
bulkAction (type:Boolean)	Yes	TRUE – indicates this request is for a bulk action (if specified then bulkDetails must be populated, and value in SIM: deviceNumber must be set to "BULK" FALSE – indicates this request is a single transaction
m2mBulkDetails	No	
m2mBulkDetail (type: m2MBulkDetailImpl)	No	See M2M Bulk Details.
deliveryDetails	Yes	List of Delivery Destinations. Must contain at least 1, but no more than 5 entries.
deliveryDetail (type: deliveryDestinationImpl)	Yes	Delivery Detail See Delivery Destination
mtFilters	Yes	List of MT Filters. May contain up to 10 entries.
mtFilter (type: mtFilterImpl)	Yes	See MT Filter.

BETA

14.2.12 OpenPort Account

The following object definition represents the OpenPort Account object (openPortSubscriberAccountImpl).

Element/Field	Required?	Description
openPortSubscriberAccount (type:openPortSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
commitmentStart (type:string)	No	This is the date the commitment obligation started for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
commitmentEnd (type:string)	No	This is the date the commitment obligation ends for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
plan (type: openPortPlanImpl)	Yes	See OpenPort Plan.
subscriberAccountMetadata (type: openPortSubscriberAccountMetadataImpl)	No	Provider reference data for the account.
spReference (type:string)	No	This is a freeform field for providers to enter reference data in regards to the SIM, for example a PO number, or customer code. This field will be searchable in accountSearch method.

BETA

Element/Field	Required?	Description
vesselName (type:string)	No	This is a freeform field for providers to enter reference data in regards to the vessel for this OpenPort Device. This field will be searchable in accountSearch method.
Sim (type:deviceSimImpl)	Yes	See SIM. While the entire object is included on the account, activations and updates only require the device number to be specified (however, the whole object is acceptable). The object will be populated on calls to Get Subscriber Account. There is a web service that allows AJAX validation of this field. See Validate Device String.
Imei (type:string)	No	International Mobile Equipment Identity This is the number that represents the physical hardware device. It is not required. NOTE: If SIM lock is ever enabled, this will be required to turn on the feature.
Ip (type:deviceIpImpl)	Yes	See Ip. While the entire object is included on the account, activations and updates only require the device number to be specified (however, the whole object is acceptable). The object will be populated on calls to Get Subscriber Account. IWS will also accept an empty string as the device number, which will trigger the auto-assign capability.

BETA

Element/Field	Required?	Description
publicIP (type:string)	No	This is the static public IP to be utilized if public ip network has been selected.
vpnId (type: string)	No	See VPN. For a selection of available VPN's, use Get Devices. This field should be set to the ID of the desired VPN. IWS also accepts a 0 which results in the use of the "Default – No VPN" option.
phoneLines	Yes	List of Phone Lines. Must contain 3 entries.
phoneLine(type: OpenPortDeviceNumberImpl)	Yes	See OpenPort Phone Number. While the entire object is included on the account, activations and updates only require the device number and voice channel options to be specified (however, the whole object is acceptable). The object will be populated on calls to Get Subscriber Account. IWS will also accept an empty string as the device number, which will trigger the auto-assign capability. The MSISDN specified should belong to the same network as the SIM device above.
scratchCardsRestricted (type:boolean)	Yes	true – This will limit scratch card usage to cards that are owned by the same service provider as the OpenPort device itself. false - This will allow any Iridium OpenPort scratch card to be used on the account.

BETA

14.2.13 Paging Account

The following object definition represents the Paging Account object (pagingSubscriberAccountImpl).

Element/Field	Required?	Description
pagingSubscriberAccount (type: pagingSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
commitmentStart (type:string)	No	This is the date the commitment obligation started for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
commitmentEnd (type:string)	No	This is the date the commitment obligation ends for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
plan (type: pagingPlanImpl)	Yes	See Paging Plan.
Capcode (type: deviceCapcodeImpl)	Yes	See Capcode. Note: This must be filled in with "BULK" if using bulk activation option.
isdna (type: deviceNumberImpl)	Yes	See Generic Number. If left empty IWS will auto generate an ISDNA number.
mtd (type: String)	No	This is the MTD serial number.
mdas	Yes	Message Delivery area. An array. This is list of up to 3 MDAs entered for activation. .
mda (type: String)	Yes	Message Delivery area. Minimum of 1 mda and maximum of 3 mda added to mdas object.

BETA

Element/Field	Required?	Description
subscriberAccountMetadata (type: absSubscriberAccountMetadataImpl)	No	
spReference (type: string)	No	Provider reference value for the account.
followMePagingNumber (type: string)	No	Used if follow me paging is selected and a service plan option.

14.2.14 Push-To-Talk Device Account

The following object definition represents the Push-To-Talk Device Account object (pttDeviceSubscriberAccountImpl).

Element/Field	Required?	Description
pttDeviceSubscriberAccount (type: pttDeviceSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
commitmentStart (type:string)	No	This is the date the commitment obligation started for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
commitmentEnd (type:string)	No	This is the date the commitment obligation ends for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
subscriptionOrgId (type: string)	No	This is the ID linking the device to a subscriber organization.
plan	Yes	See Push-To-Talk Device Plan.

BETA

Element/Field	Required?	Description
(type: pttDevicePlanImpl)		
imei (type: string)	Yes	IMEI for the device account. This number will be validated for TAC/FAC and 15 digit length. The IMEI for Push-To-Talk is not a device maintained in resource inventory,
bulkAction (type: Boolean)	Yes	TRUE – indicates this request is for a bulk action (if specified then bulkDetails must be populated, FALSE – indicates this request is a single transaction
pttDeviceBulkDetails	No	
subscriberAccountMetadata (type: absSubscriberAccountMetadataImpl)	No	
spReference (type: string)	No	Provider reference value for the account.

14.2.15 Push-To-Talk Talkgroup Account

The following object definition represents the Push-To-Talk Talkgroup Account object (pttTalkgroupSubscriberAccountImpl).

Element/Field	Required?	Description
pttTalkGroupSubscriberAccount (type: pttTalkGroupSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
commitmentStart (type:string)	No	This is the date the commitment obligation started for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.

BETA

Element/Field	Required?	Description
commitmentEnd (type:string)	No	This is the date the commitment obligation ends for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
Name (type: string)	No	This is the name for the talkgroup. This will be maintained at the controller, and is for display purposes only through IWS.
talkgroupId (type: string)	No	This is the Talkgroup ID assigned by the PTT Controller.
subscriptionOrgId (type: string)	No	This is the ID linking the talkgroup to a subscriber organization.
plan (type: pttTalkGroupPlanImpl)	Yes	See Push-To-Talk Talkgroup Plan.
subscriberAccountMetadata (type: absSubscriberAccountMetadataImpl)	No	
spReference (type: string)	No	Provider reference value for the account.

BETA

14.2.16 RUDICS Group

The following object definition represents the RUDICS Group object (rudicsGroupImpl).

Element/Field	Required?	Description
rudicsGroup (type: rudicsGroupImpl)	No	
id (type:string)	Yes	RUDICS group ID
name (type: string)	Yes	RUDICS group name
description (type: string)	No	
rudicsType (type: rudicsTypeEnum)	No	See rudicsTypeEnum
group (type:string)	Yes	RUDICS group identifier. Similar to name.

14.2.17 SBD Account

The following object definition represents the SBD Account object (sbdSubscriberAccountImpl).

Element/Field	Required?	Description
sbdSubscriberAccount (type: sbdSubscriberAccountImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
plan (type: sbdPlanImpl)	Yes	See SBD Plan.
subscriberAccountMetadata (type: sbdSubscriberAccountMetadataImpl)	No	Provider reference data for the account.
spReference (type:string)	No	This is a freeform field for providers to enter reference data in regards to the SIM, for example a PO number, or customer code. This field will be searchable in accountSearch method.
Imei (type:string)	Yes	Must be a valid IMEI for an SBD device.

BETA

Element/Field	Required?	Description
		At a very basic level, it should be 15-digit number starting with a 3 and ending with a 0. There is a web service that allows AJAX validation of this field. See Validate Device String.
homeGateway (type:homeGatewayEnum)	No	This value will be present for customers with the capability to override default home gateway on SIM-based IMEIs on SBD subscriber accounts. This is not available for non SIM-based IMEIs, or for customers without home gateway override privileges.
bulkAction (type:Boolean)	Yes	TRUE – indicates this request is for a bulk action (if specified then bulkDetails must be populated, and value in SIM: deviceNumber must be set to "BULK" FALSE – indicates this request is a single transaction
sbdBulkDetails	No	
sbdBulkDetail (type: sbdBulkDetailImpl)	No	See SBD Bulk Details.
deliveryDetails	Yes	List of Delivery Destinations. Must contain at least 1, but no more than 5 entries.
deliveryDetail (type: deliveryDestinationImpl)	Yes	List of Delivery Destinations. Must contain at least 1, but no more than 5 entries. See Delivery Destination.
mtFilters	Yes	List of MT Filters. May contain up to 10 entries.
mtFilter (type: mtFilterImpl)	Yes	See MT Filter.

BETA

14.2.18 Telephony Base Account Object

The following object definition represents the base Telephony Account object (telephonySubscriberAccountBaseImpl).

Element/Field	Required?	Description
telephonySubscriberAccount (type: telephonySubscriberAccountBaseImpl)	No	
absSubscriberAccountImpl	Yes	Base account object
commitmentStart (type:string)	No	This is the date the commitment obligation started for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
commitmentEnd (type:string)	No	This is the date the commitment obligation ends for the selections purchased. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z' which is ISO8601 standard format. It cannot be modified directly, but it will be returned on searches such as Get Subscriber Account.
plan (type: telephonyPlanImpl)	Yes	See Telephony Plan.
sim (type: deviceSimImpl)	Yes	See SIM. Note: This must be filled in with “BULK” if using bulk activation option.
imei (type: string)	No	International Mobile Equipment Identity This is the number that represents the physical hardware device. Required for LBS plans.

BETA

Element/Field	Required?	Description
msisdn (type: deviceNumberImpl)	Yes	See Telephony Number. Note: This must be filled in with “BULK” if using bulk activation option.
msisdnb (type: deviceNumberImpl)	No	See Telephony Number.
msisdnc (type: deviceNumberImpl)	Yes	See Telephony Number. Note: This must be filled in with “BULK” if using bulk activation option.
enableLocalNumber (type: string)	No	TRUE – Turn on local number (singlestage) dialing for this line. FALSE - Do not turn on local number dialing for this line.
localCountry (type: string)	No	Id for the country/region to select the local number resource. See Get Local Number Regions
plusNumber (type: deviceNumberImpl)	Yes	See Telephony Number. Note: This must be filled in with “BULK” if using bulk activation option.
bulkAction (type: Boolean)	Yes	TRUE – indicates this request is for a bulk action (if specified then bulkDetails must be populated, and value in SIM: deviceNumber, msisdn:deviceNumber, and msisdnc:deviceNumber must be set to “BULK” FALSE – indicates this request is a single transaction
telephonyBulkDetails	No	
telephonyBulkDetail (type: telephonyBulkDetailImpl)	No	See Telephony Bulk details.
subscriberAccountMetadata (type: telephonySubscriberAccountMetadataImpl)	No	
spReference (type: String)	No	Provider reference value for the account.
twoStageVoice (type: twoStageImpl)	Yes	See Two Stage
twoStageData (type: twoStageImpl)	Yes	See Two Stage

BETA

14.2.19 Telephony Account

The following object definition represents the Telephony Account object (telephonySubscriberAccountImpl).

Element/Field	Required?	Description
telephonySubscriberAccount (type: telephonySubscriberAccountImpl)	No	
telephonySubscriberAccount (type: telephonySubscriberAccountBaseImpl)	No	See Telephony Base Account Object
activatePrepaid (type: Boolean)	Yes	TRUE – indicates this request is intended to create an account on the prepaid system for the SIM in question. If TRUE the prepaidPlan object is required. FALSE – indicates this request does not intend to activate a prepaid account object, however prepaid plan object may be included if already activated.
prepaidPlan (type: prePaidPlanImpl)	No	See Telephony Prepaid Plan.
activateCrew (type: Boolean)	Yes	TRUE – indicates this request is intended to create a crew calling account on the prepaid system for the SIM in question. If TRUE the crew Plan object is required. FALSE – indicates this request does not intend to activate a crew calling account object, however crew plan object may be included if already activated.
crewPlan (type: crewPlanImpl)	No	See Telephony Crew Plan.

14.2.20 Prepaid Account Detail

This is the object that represents the prepaid account detail object. (prepaidAccountDetailImpl)

Element/Field	Required?	Description
msisdn (type: string)	Yes	Represents the phone number for VOICE calls. MSISDN is only used for prepaid.
iccid (type: string)	No	This is the number representing a SIM card. Here are some examples: 8988169703000007348 8988169777000016038

BETA

Element/Field	Required?	Description
		The bold numbers in the middle of the SIM number represent the network that the device belongs to. This same series of numbers is present in the MSISDNs that are activated on the account. When not applicable, it will be populated with "NA".
createdBy (type: string)	Yes	Prepaid user that created account
createdDate (type: string)	Yes	Created date for prepaid account Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
expirationDate (type: string)	No	Expiration date for prepaid account. Format for date string: yyyy-MM-dd'T'HH:mm:ss'Z'
lockState (type: string)	No	
pinNumber (type: string)	No	
currentBalance (type: long)	No	The current balance of units available of prepaid account.
servicePlan (type: string)	Yes	Prepaid plan value.
restrictScratchCards (type: boolean)	No	True/False Is restrict flag set
spAlert (type: boolean)	No	Alert service provider for balance expiration. Expiration and Balance alert warnings TRUE – Enabled FALSE – Disabled Value Defaults to FALSE
language (type:prepaidLanguageTypeEnum)	Yes	Announcement language. See Prepaid Language Type Enum.
accountNo (type: string)	No	Account Number for SIM linked to this account.
isuFlag (type: boolean)	Yes	True/False Is ISU Network bypass flag Value Defaults to FALSE

BETA

Element/Field	Required?	Description
status (type: string)	Yes	Status of prepaid account.
unitExpiration (type: string)	No	Amount of units to expire in next expiration for outdated vouchers.
unitExpirationDate (type: string)	No	Date of next unit expiration for outdated vouchers.
unitExpirationNotice (type: string)	No	Notice providing details about next automatic unit expiry for outdated vouchers.

14.2.21 Subscriber Organization

This is the object that represents the Push-To-Talk subscriber Organization. (subscriberOrganizationImpl). While this object is not an actual account object, it has been included here for reference purposes.

Element/Field	Required?	Description
id (type:string)	Yes	Subscriber Organization ID
refid(type:string)	Yes	Reference ID from controller
submarketId (type:string)	Yes	Submarket ID for the subscriber Organization
organizationName (type:string)	Yes	Name for Subscriber Organization
description (type:string)	No	Optional field for user desired Description
status (type: statusEnum)	No	
createDate (type:string)	No	

14.2.22 Vessel

This is the object that represents a vessel or ship for maritime safety. (vesselImpl)

Element/Field	Required?	Description
id (type:string)	Yes	Vessel ID
vesselName (type:string)	Yes	Vessel Name
vesselType (type: VesselTypeEnum)	Yes	See vesselTypeEnum
fleetName (type:string)	No	Associated fleet name.
imo (type:string)	Yes	International Maritime Organization number. Can be set to EXEMPT if data not applicable

BETA

mmsi (type: statusEnum)	Yes	Maritime Mobile Service Identity
callSign (type:string)	No	Radio call sign
shipRegistry (type:string)	No	Two character ISO code country identifier
vesselLinks	No	An array of contract links to the vessel
vesselLink(type: VesselLinkImpl)	No	See Vessel Link

14.2.23 Vessel Link

A vessel link object is the connection between a vessel and the associated contracts to the vessel. (vesselLinkImpl)

Element/Field	**Required?**	**Description**
subscriberAccountNumber (type:string)	Yes	Contract ID
type (type: vesselLinkTypeEnum)	Yes	See vesselLinkTypeEnum
serviceType (type: serviceTypeEnum)	Yes	See serviceTypeEnum
sim (type: string)	Yes	Contract SIM
imei (type: string)	No	Contract IMEI
createdDate (type: string)	Yes	Date the contract was created
modifiedDate (type: string)	Yes	Date the contract was last modified

BETA

14.3 Plan

Plans are used to specify the elements of a device that either have pricing tied directly to them or can affect other selections you may have. These objects are part of the account object and will also be passed into the bundle-related requests (like Get Voice Bundles). Those requests trigger the execution of Iridium's business rules to determine what the valid selections are based on what has already been specified.

14.3.1 Base Plan Object

absPlanImpl describes the base plan object. This is used in requests that do not need the full plan object.

Element/Field	Required?	Description
promo (type: string)	Yes	One of the values available from Get Promo Bundles.
demoAndTrial (type: string)	Yes	One of the values available from Get Demo And Trial Bundles.
accountPoolingGroup (type: string)	No	0 or one of the values available from Get Account Pooling Groups. Will default to 0.

14.3.2 Burst Device Plan

burstDevicePlanImpl describes the plan object for Burst accounts.

Element/Field	Required?	Description
absPlanImpl	Yes	
burstDeviceBundleId (type:string)	Yes	One of the values available from Get Burst Device Bundles.

14.3.3 Burst Service Plan

burstServiceSubAccountPlanImpl describes the plan object for Burst Service accounts.

Element/Field	Required?	Description
absPlanImpl	Yes	
burstServiceBundleId (type:string)	Yes	One of the values available from Get Burst Subscription Names.

BETA

14.3.4 Certus Plan

certusPlanImpl describes the plan object for Certus accounts.

Element/Field	Required?	Description
subMarketId (type: string)	Yes	This is the internal sub-market ID that identifies the sub market that the activation will use.
servicePlanId (type: string)	Yes	This is the servicePlan internal Id that identifies with plan the activation will use.
servicePlanOptions	Yes	Array of options for the service plan.
servicePlanOptionId (type: string)	Yes	This is the individual planOption ID available. This could be one to many. See Service Plan Options.

14.3.5 M2M+ Plan

m2mPlanImpl describes the plan object for M2M+ accounts.

Element/Field	Required?	Description
absPlanImpl	Yes	
m2mBundleId (type: string)	No	One of the values available from Get M2M Bundles.
rudicsGroup (type: string)	No	0 or one of the values available from Get RUDICS Groups. Will default to 0.
lritFlagstate (type:string)	No	Long Range Identification and Tracking. If an LRIT plan is selected (i.e., LRIT A or LRIT B), a valid 3-character alphanumeric string is required. See Appendix A for a list of vessel flag codes, Otherwise, empty string is acceptable.
ringAlertsFlag (type:boolean)	No	TRUE – Enabled FALSE - Disabled Will default to false.

14.3.6 OpenPort Plan

openPortPlanImpl describes the plan object for OpenPort accounts.

BETA

Element/Field	Required?	Description
absPlanImpl	Yes	
paymentModel (type:paymentModelTypeEnum)	Yes	See Payment Models. This is always POST_PAID for OpenPort.
voiceBundleId (type:string)	Yes	One of the values available from Get Voice Bundles.
dataBundleId (type: string)	Yes	One of the values available from Get Data Bundles.
dataSpeedBundleId (type: string)	Yes	One of the values available from Get Data Speeds.
vcsBundleId (type: string)	Yes	One of the values available from Get VCS Bundles.
commitmentPeriodBundleId (type: string)	Yes	One of the values available from Get Commitment Periods.
languageBundleId (type: string)	Yes	One of the values available from Get Languages.
simLock (type:string)	No	true – Locks the OpenPort unit to the SIM card specified at provisioning for the duration of the commitment period. false – Do not lock the hardware device. Note: If true, this is associated by IMEI field listed below. A minimum of 1 year commitment must be sent in request. Empty IMEI and “None” commitment will result in a failed request.

BETA

14.3.7 Paging Plan

pagingPlanImpl describes the plan object for Paging accounts.

Element/Field	Required?	Description
pagingPlanBaseImpl	Yes	Paging Base Plan object.
language (type: telephonyLanguageTypeEnum)	Yes	Language to be used for voicemail prompts, must be provided.
subMarketId (type: string)	Yes	This is the internal sub-market ID that identifies the sub market that the activation will use.
servicePlanId (type: string)	Yes	This is the servicePlan internal Id that identifies with plan the activation will use.
servicePlanOptions	Yes	Array of options for the service plan.
servicePlanOptionId (type: string)	Yes	This is the individual planOption ID available. This could be one to many. See Service Plan Options.

14.3.8 Push-To-Talk Device Plan

pttDevicePlanImpl describes the plan object for Push-To-Talk Device accounts.

Element/Field	Required?	Description
pttDevicePlanImpl	Yes	Push-To-Talk Device Base Plan object.
subMarketId (type: string)	Yes	This is the internal sub-market ID that identifies the sub market that the activation will use.
servicePlanId (type: string)	Yes	This is the servicePlan internal Id that identifies with plan the activation will use.
servicePlanOptions	Yes	Array of options for the service plan.
servicePlanOptionId (type: string)	Yes	This is the individual planOption ID available. This could be one to many. See Service Plan Options.

BETA

14.3.9 Push-To-Talk Talkgroup Plan

pttTalkgroupPlanImpl describes the plan object for Push-To-Talk Talkgroup accounts.

Element/Field	Required?	Description
pttTalkgroupPlanImpl	Yes	Push-To-Talk Talkgroup Base Plan object.
subMarketId (type: string)	Yes	This is the internal sub-market ID that identifies the sub market that the activation will use.
servicePlanId (type: string)	Yes	This is the servicePlan internal Id that identifies with plan the activation will use.
servicePlanOptions	Yes	Array of options for the service plan.
servicePlanOptionId (type: string)	Yes	This is the individual planOption ID available. This could be one to many. See Service Plan Options.

14.3.10 SBD Plan

sbdPlanImpl describes the plan object for SBD accounts.

Element/Field	Required?	Description
absPlanImpl	Yes	
sbdBundleId (type: string)	Yes	One of the values available from Get SBD Bundles.
lritFlagstate (type:string)	Yes	Long Range Identification and Tracking. If an LRIT plan is selected (i.e., LRIT A or LRIT B), a valid 3-character alphanumeric string is required. See Appendix A for a list of vessel flag codes, Otherwise, empty string is acceptable.
ringAlertsFlag (type:boolean)	No	TRUE – Enabled FALSE - Disabled Will default to false.

BETA

14.3.11 Telephony Plan

telephonyPlanImpl describes the plan object for Telephony accounts.

Element/Field	Required?	Description
telephonyPlan (type: telephonyPlanBaseImpl)	Yes	Telephony Base Plan Object.
language (type: languageTypeEnum)	Yes	See LanguageTypeEnum
subMarketId (type: string)	Yes	This is the internal sub-market ID that identifies the sub market that the activation will use.
servicePlanId (type: string)	Yes	This is the servicePlan internal Id that identifies with plan the activation will use.
servicePlanOptions	Yes	Array of options for the service plan.
servicePlanOptionId (type: string)	Yes	This is the individual planOption ID available. This could be one to many. See Service Plan Options.
rudicsGroup(type: string)	No	0 or one of the values available from Get RUDICS Groups. Will default to 0.

14.3.12 Crew Plan

CrewPlanImpl describes the plan object for crew accounts.

Element/Field	Required?	Description
prepaidAccountDetail (type: prepaidAccountDetailImpl)	Yes	See Prepaid Account Detail

14.3.13 Prepaid Plan

PrePaidPlanImpl describes the plan object for prepaid accounts

Element/Field	Required?	Description
prepaidAccountDetail (type: prepaidAccountDetailImpl)	Yes	See Prepaid Account Detail.
prePaidVoucher(type: prePaidVoucherImpl)	Yes	See Prepaid Voucher. This can be empty.

14.4 Bulk Details

The following objects are utilized when performing bulk actions.

BETA

14.4.1 M2M+ Bulk Details

This object is utilized to apply bulk actions to activation and update requests.

Element/Field	Required?	Description
m2mBulkDetails	Yes	
m2MBulkDetail(type: m2MBulkDetailImpl)	Yes	List of device details to be affected by bulk action request.
sim (type: deviceSimImpl)	Yes	See SIM.
imei (type: string)	No	Specify IMEI of devices to be affected by the bulk action.

14.4.2 SBD Bulk Details

This object is utilized to apply bulk actions to activation and update requests.

Element/Field	Required?	Description
sbdBulkDetails	Yes	
sbdBulkDetail (type: sbdBulkDetailImpl)	Yes	List of device details to be affected by bulk action request.
imei (type: string)	No	Specify IMEI of devices to be activated or updated.

14.4.3 Telephony Bulk Details

This object is utilized to apply bulk actions to Telephony activation and update requests.

Element/Field	Required?	Description
telephonyBulkDetails	Yes	
telephonyBulkDetail (type: telephonyBulkDetailImpl)	Yes	List of device details to be affected by bulk action request.
sim (type: string)	Yes	This is the number representing a SIM card. Here are some examples: 8988169703000007348 8988169777000016038
imei (type: string)	Yes	International Mobile Equipment Identity This is the number that represents the physical hardware device. Required for LBS plans.

BETA

14.4.4 Push-To-Talk Bulk Details

This object is utilized to apply bulk actions to Push To Talk device activation and update requests.

Element/Field	Required?	Description
pttDeviceBulkDetails	Yes	
pttDeviceBulkDetail (type: pttDeviceBulkDetailImpl)	Yes	List of device details to be affected by bulk action request.
imei (type: string)	Yes	International Mobile Equipment Identity This is the number that represents the physical hardware device.

14.5 Bulk Plan Update

The following objects are utilized when performing bulk transition actions.

14.5.1 M2M+ Bulk Plan Update

This object is utilized to apply bulk actions to activation and update requests. (m2MBulkPlanUpdateImpl)

Element/Field	Required?	Description
currentPlan (type: m2mBulkPlanImpl)	Yes	
updatePlan (type: m2mBulkPlanImpl)	Yes	
m2mBulkDetails	Yes	See M2M Bulk Details.

14.5.2 Push-To-Talk Device Bulk Plan Update

This object is utilized to apply bulk actions to activation and update requests. (pttDeviceBulkPlanUpdateImpl)

Element/Field	Required?	Description
submarketId (type: string)	Yes	
currentServicePlan (type: pttDeviceBulkBaseImpl)	Yes	
updateServicePlan (type: pttDeviceBulkPlanImpl)	Yes	
pttDeviceBulkDetails	Yes	See Push-To-Talk Device Bulk Details.

14.5.3 SBD Bulk Plan Update

This object is utilized to apply bulk actions to activation and update requests. (sbdBulkPlanUpdateImpl)

BETA

Element/Field	Required?	Description
currentPlan (type: sbdBulkPlanImpl)	Yes	
updatePlan (type: sbdBulkPlanImpl)	Yes	
sbdBulkDetails	Yes	See SBD Bulk Details.

14.5.4 Telephony Bulk Plan Update

This object is utilized to apply bulk actions to activation and update requests. (telephonyBulkPlanUpdateImpl)

Element/Field	Required?	Description
submarketId (type: string)	Yes	
currentServicePlan (type: telephonyBulkBaseImpl)	Yes	
updateServicePlan (type: telephonyBulkPlanImpl)	Yes	
telephonyBulkDetails	Yes	See Telephony Bulk Details.

BETA

14.5.5 SBD Bulk Plan

sbdBulkPlanImpl describes the bulk plan object for SBD accounts.

Element/Field	Required?	Description
sbdBundleId (type: string)	No	One of the values available from Get SBD Bundles.
accountPoolingGroup (type: string)	No	0 or one of the values available from Get Account Pooling Groups. Will default to 0.

14.5.6 M2M+ Bulk Plan

m2MBulkPlanImpl describes the bulk plan object for M2M+ accounts.

Element/Field	Required?	Description
m2mBundleId (type: string)	No	One of the values available from Get M2M Bundles.
accountPoolingGroup (type: string)	No	0 or one of the values available from Get Account Pooling Groups. Will default to 0.

14.5.7 Push-To-Talk Device Bulk Plan

pttDeviceBulkPlanImpl describes the bulk plan object for Push-To-Talk accounts.

Element/Field	Required?	Description
type: pttDeviceBulkBaseImpl	No	
submarketId (type:string)	No	A valid submarket ID which will apply to all sims for this action.
servicePlanOptions	No	An array of service Plan Options to apply settings for the bulk request.
servicePlanOptionId (type:string)	No	Individual service options to be purchased with the accounts.

14.5.8 Push-To-Talk Bulk Base

pttDeviceBulkBaseImpl describes the bulk base object for Push-To-Talk Device updates.

Element/Field	Required?	Description
servicePlanId (type: string)	No	One of the values available from Get ServicePlans.

14.5.9 Telephony Bulk Plan

telephonyBulkPlanImpl describes the bulk plan object for Telephony accounts.

BETA

Element/Field	Required?	Description
type: telephonyBulkBaseImpl	No	
submarketId (type:string)	No	A valid submarket ID which will apply to all sims for this action.
servicePlanOptions	No	An array of service Plan Options to apply settings for the bulk request.
servicePlanOptionId (type:string)	No	Individual service options to be purchased with the accounts.
rudicsGroups	No	
rudicsGroup (type: rudicsGroupImpl)	No	See RUDICS Group
twoStageVoice (type: twoStageImpl)	No	Two stage voice settings
twoStageData (type: twoStageImpl)	No	Two stage Data settings
enableLocalNumber (type:boolean)	No	TRUE – Turn on local number (singlestage) dialing for this line. FALSE - Do not turn on local number dialing for this line. **For Future Use**, Set to false until local number capabilities are expanded for Telephony
localCountry (type:string)	No	Id for the country/region to select the local number resource. **For Future Use,** Leave blank until local number capabilities are expanded for Telephony See Get Local Number Regions

BETA

14.5.10 Telephony Bulk Base

telephonyBulkBaseImpl describes the bulk base object for Telephony updates.

Element/Field	Required?	Description
servicePlanId (type: string)	No	One of the values available from Get ServicePlans.
accountPoolingGroup (type: string)	No	0 or one of the values available from Get Account Pooling Groups. Will default to 0.

14.6 Device / Number Objects

14.6.1 Capcode

This is the object that represents the Capcode enabling service to the device. (deviceCapcodeImpl)

Element/Field	Required?	Description
deviceNumber (type: string)	Yes	A unique number assigned to each pager unit, which is used to identify the paging unit on the network and may be used for billing purposes.
deviceId (type: string)	No	This is a unique numerical identifier assigned to the device by the billing system. It is provided for information purposes only.
subMarketId (type: string)	Yes	This is the internal sub-market ID that identifies the sub market that the activation will use.
network (type:string)	Yes	This is the name of the network that the Capcode number belongs to.

14.6.2 Certus Voice Line

This is the object that represents the voice line details including phone number/MSISDN assigned to a voice channel on the Certus device. (certusDeviceNumberImpl)

Element/Field	Required?	Description
deviceNumber (type:string)	Yes	MSISDN for Certus voice line. See Generic Number
deviceId (type:string)	Yes	Internal ID for device resource
subMarketId (type: string)	Yes	This is the internal sub-market ID that identifies the sub market that the activation will use.

BETA

Element/Field	Required?	Description
network (type: string)	Yes	
linenum (type: int)	Yes	Line number associated to the voice line.
lineType (type: certusLineTypeEnum)	Yes	Primary line type. See Certus Line Type
lineQuality (type: lineQualityEnum)	Yes	Line quality. See Line Quality
voiceMail (type:boolean)	Yes	TRUE – Turn on voicemail access for this line. FALSE - Do not turn on voicemail access for this line. Not allowed on Prepaid or Safety lines.
captainPinEnable (type:boolean)	Yes	TRUE – Turn on a captain PIN access for this line. FALSE - Do not turn on a captain PIN for this line. Not allowed on Prepaid or Safety lines..
captainPin (type:string)	Yes	This is a 4-digit PIN number that should be set if the captain pin feature is enabled. It will be required to complete an outgoing call, if enabled. Not allowed on Prepaid or Safety lines..
twoStageEnable (type:boolean)	Yes	TRUE – Turn on two stage dialing for this line. FALSE - Do not turn on two stage dialing for this line. Not allowed on Prepaid or Safety lines.
twoStagePin (type:string)	No	When two stage dialing is enabled, this should be set to one of the following: A 4-digit PIN number Empty string NOPIN When two stage dialing is disabled, this should be set to one of the following: Empty string DIS

BETA

Element/Field	Required?	Description
enableLocalNumber (type:boolean)	Yes	TRUE – Turn on local number (singlestage) dialing for this line. FALSE - Do not turn on local number dialing for this line. Not allowed on Prepaid or Safety lines.
localCountry (type:string)	No	Id for the country/region to select the local number resource. See Get Local Number Regions
localNumber (type: localNumImpl)	No	
localCountryNumber (type: string)	No	The local number is assigned by IWS at the time it is enabled; thus, it should be an empty string on requests for activations or updates. It will be populated on calls to getSubscriberAccount and on the responses to activations and updates.
numberModel (type: string)	No	The number model will indicate which type of local number has been assigned.
language (type:string)	No	TRUE – Turn on local number (singlestage) dialing for this line. FALSE - Do not turn on local number dialing for this line. Not allowed on Prepaid or Safety lines.
primarySafety (type:string)	No	TRUE – voice line is safety type and is designated as the primary safety line for the account. FALSE - voice line is safety type and is not designated as the primary safety line for the account. Only permitted if safety enabled is TRUE.

14.6.3 Secondary Data Service

Secondary Data Service is the service definition of a secondary data service which is utilized by Certus accounts. This is the service object that will be associated to Certus contracts.

The service plan object will contain the following elements:

Element/Field	Required?	Description
secondaryDataService (type: secondaryDataServiceImpl)	Yes	Secondary data service object

BETA

Element/Field	Required?	Description
id (type: string)	No	Internal system identifier for the secondary data service. This is to be used when adding a secondary data service to an activation or update request. For creation of new Secondary Data Service this value should not be provided but will be necessary on update as the main reference value.
serviceName (type: string)	Yes	Name of the secondary data service
billingId (type: secondaryDataServiceBillingIdEnum)	Yes	indicates vlan position of secondary data service
subnetValue (type: string)	Yes	Base IP address of Gateway Subnet for the service. IPV4 address in dotted quad format.
subnetMask (type: string)	Yes	Mask for Gateway Subnet for the service in dotted quad format.
defaultPepEnabled (type Boolean)	Yes	0, 2 = Disabled; 1 = Enabled;
defaultHdrCompEnabled (type Boolean)	Yes	0, 2 = Disabled; 1 = Enabled;
defaultPayloadCompEnabled (type Boolean)	Yes	0, 2 = Disabled; 1 = Enabled;
priority (type: certusSecondaryDataServicePriorityEnum)	Yes	EXPEDITED - SAPI_0 NORMAL - SAPI_1
sdfRatingId (type:string)	Yes	3 character value used for data extract

14.6.4 Secondary Data Services Map

Secondary Data Service Map shows the connection between a Secondary Data Service and a Service Plan.

The service plan object will contain the following elements:

Element/Field	Required?	Description
secondaryDataServicesMap (type: secondaryDataServicesMapImpl)	Yes	Secondary data services map object

BETA

Element/Field	Required?	Description
servicePlanId (type: string)	Yes	Service Plan ID that the Secondary Data Service is associated to. The Secondary Data Service will be available on that plan for activation or update request.
secondaryDataServices	Yes	List of Secondary Data Services
secondaryDataService (type: secondaryDataServiceImpl)	Yes	See Secondary Data Service

14.6.5 SIM

This is the object that represents the SIM card enabling service to the device. (deviceSimImpl)

Element/Field	Required?	Description
deviceNumber (type:string)	Yes	This is the number representing a SIM card. Here are some examples: 8988169703000007348 8988169777000016038 The bold numbers in the middle of the SIM number represent the network that the device belongs to. This same series of numbers is present in the MSISDNs that are activated on the account.
deviceId (type: string)	No	This is a unique numerical identifier assigned to the device by the billing system. It is provided for information purposes only.
imsi (type:string)	No	International Mobile Subscriber Identity See the following page for details on what an IMSI is: http://en.wikipedia.org/wiki/International_Mobile_Subscriber_Identity It is provided for information purposes only.
subMarketId (type: string)	No	This is the internal sub-market ID that identifies the sub market that the activation will use.
submarketType (type: string)	No	This is the internal sub-market type that identifies the sub market that the activation will use.

BETA

Element/Field	Required?	Description
network (type:string)	No	This is the name of the network that the SIM number belongs to. It is provided for information purposes only.

14.6.6 IMEI

This is the object that represents the IMEI enabling service to the device. (deviceImeiImpl)

Element/Field	Required?	Description
deviceNumber (type:string)	Yes	This is the number representing an IMEI. Here are some examples: 300215070014245 300215079999999
deviceId (type: string)	No	This is a unique numerical identifier assigned to the device by the billing system. It is provided for information purposes only.

14.6.7 Generic Number

This is the object that represents the generic device number (deviceNumberImpl)

Element/Field	Required?	Description
deviceNumber (type:string)	Yes	This number represents the identifier for a phone number. In most cases this refers to a MSISDN, but could also be an IP address or paging capcode.
deviceId (type: string)	No	This is a unique numerical identifier assigned to the device by the billing system. It is provided for information purposes only.
numberType (type: deviceNumberTypeEnum)	Yes	This is the object that represents the VPN that the IP has access to.
localNumberModel (type: string)	No	Used for PlusNumber object to indicate model of the device.
subMarketId (type: string)	Yes	This is the internal sub-market ID that identifies the sub market that the activation will use.

BETA

Element/Field	Required?	Description
submarketType (type: string)	No	This is the internal sub-market type that identifies the sub market that the activation will use.
network (type:string)	Yes	This is the name of the network that the number belongs to.
State (type:deviceStatusTypeEnum)	No	Status of device See deviceStatusTypeEnum

14.6.8 OpenPort Phone Number

This is the object that represents the phone number/MSISDN assigned to a single voice channel on the OpenPort device. (openPortDeviceNumberImpl)

Element/Field	Required?	Description
deviceNumber (type:deviceNumberImpl)	Yes	MSISDN for OpenPort voice line. See Generic Number
voiceChannel (type:int)	Yes	1, 2, or 3 This represents the channel on the OpenPort hardware that will be used for this particular number.
lineType (type:lineTypeEnum)	Yes	Phone lines can have one of two types of functionality: Crew and Standard STANDARD -This line type supports both outbound and inbound calls and can be configured to support voice mail, security PIN, 2-Stage PIN and Plus 1 features. CREW - This line type is restricted to prepaid use.
voiceMail (type:string)	Yes	TRUE – Turn on voicemail access for this line. FALSE - Do not turn on voicemail access for this line. Not allowed on CREW lines.

BETA

Element/Field	Required?	Description
captainPinEnable (type:boolean)	Yes	TRUE – Turn on a captain PIN access for this line. FALSE - Do not turn on a captain PIN for this line. Not allowed on CREW lines.
captainPin (type:string)	Yes	This is a 4-digit PIN number that should be set if the captain pin feature is enabled. It will be required to complete an outgoing call, if enabled. Not allowed on CREW lines.
twoStageEnable (type:boolean)	Yes	TRUE – Turn on two stage dialing for this line. FALSE - Do not turn on two stage dialing for this line. Not allowed on CREW lines.
twoStagePin (type:string)	No	When two stage dialing is enabled, this should be set to one of the following: A 4-digit PIN number Empty string NOPIN When two stage dialing is disabled, this should be set to one of the following: Empty string DIS
plusOneEnable (type:boolean)	Yes	TRUE – Turn on plus one (+1) dialing for this line. Not allowed on CREW lines. FALSE - Do not turn on plus one dialing for this line. Not allowed on CREW lines.
plusOneNumber (type:string)	No	The plus one number is assigned by IWS at the time it is enabled; thus, it should be an empty string on requests for activations or updates. It will be populated on calls to getSubscriberAccount and on the responses to activations and updates.
programmableButtons	Yes	This is a list of programmable short-codes. No more than 5 may be provided.
programmableButton1 (type:string)	Yes	This is one of the 4-digit short codes. 2881 and 2882 are mandatory.

BETA

Element/Field	Required?	Description
programmableButton2 (type:string)	Yes	This is one of the 4-digit short codes. 2881 and 2882 are mandatory.
programmableButton3 (type:string)	Yes	This is one of the 4-digit short codes. 2881 and 2882 are mandatory.
programmableButton4 (type:string)	Yes	This is one of the 4-digit short codes. 2881 and 2882 are mandatory.
programmableButton5 (type:string)	Yes	This is one of the 4-digit short codes. 2881 and 2882 are mandatory.

14.6.9 IP

This is the object that represents the IP address assigned to the device. (deviceIpImpl)

Element/Field	Required?	Description
deviceNumber (type:string)	Yes	This is the number representing an IP address. Here is an example: 127.0.0.1
deviceId (type: string)	No	This is a unique numerical identifier assigned to the device by the billing system. It is provided for information purposes only.
vpn (type:deviceApnImpl)	Yes	This is the object that represents the VPN that the IP has access to. See VPN.

14.6.10 LocalCountry

This is the object that represents the Local Number Region assigned to the device. (localCountryImpl)

Element/Field	Required?	Description
deviceNumber (type:string)	Yes	This is the number representing an IP address. Here is an example: 127.0.0.1
deviceId (type: string)	No	This is a unique numerical identifier assigned to the device by the billing system. It is provided for information purposes only.
localCountry (type:string)	No	The country/region id

BETA

14.6.11 VPN

This is the object that represents the VPN that the IP address has access to. (deviceApnImpl)

Element/Field	Required?	Description
deviceNumber (type:string)	Yes	This is the number representing a VPN (Virtual Private Network). Here is an example: 127.0.0.1
deviceId (type: string)	No	This is a unique numerical identifier assigned to the device by the billing system. It is provided for information purposes only.
vpnName (type:string)	Yes	This is the name of the VPN. It can be used for display purposes.
Colotype (type:string)	No	
routingInfo (type:int)	No	Identifier for Certus data routing
streamingCapable (type:Boolean)	No	Determines if VPN supports streaming
sdfCapable (type:Boolean)	No	Determines if VPN supports secondary data services
prepaidDataCapable (type:Boolean)	No	Determines if VPN supports prepaid data
standardVoiceCapable (type:Boolean)	No	Future setting
highVoiceCapable (type:Boolean)	No	Future setting

14.6.12 Submarket

This is the object that represents the submarket for devices (submarketImpl)

Element/Field	Required?	Description
id (type: string)	Yes	
name (type: string)	Yes	The name of the actual submarket available to the user.
code (type: string)	Yes	Identifier for the submarket. This is to be used when adding a service plan to an activation or update request.
submarketType (type: submarketTypeEnum)	Yes	See Submarket Types.
servicePlans	No	This is the list of Service Plans available. This could be empty.

BETA

Element/Field	Required?	Description
servicePlan (type: absServicePlanImpl)	No	This is the service plan available. This could be empty. See Service Plan

BETA

14.7 Delivery Destinations

Delivery Destinations specify the delivery destinations an SBD or M2M device (IMEI) will upload/send data to. Valid destinations can be any combination of E-Mail, Direct IP, or other IMEIs. One delivery destination is required; however, up to 5 are supported. Wildcard characters (*) are NOT supported.

Examples of destinations (by delivery method) are shown below:

- E-mail: Tom@VendorDomain.com
- Direct IP: 168.45.34.4:10921 (NOTE: The port is required)
- Direct IP:destination.com:1234 (NOTE: The port is required)
- SBD Device: 300040100001230

Element/Field	**Required?**	**Description**
DeliveryDetail (type:deliveryDestinationImpl)	Yes	
destination (type:string)	Yes	Based on the deliveryMethod specified, this should represent the IP, email, or another SBD device you would like to send data to. See examples above for valid values.
deliveryMethod (type:deliveryMethodTypeEnum)	Yes	See Delivery Method Type.
geoDataFlag (type:string)	Yes	Represents whether the device should include geographical data when it sends data to these destinations. TRUE – Enabled FALSE - Disabled
moAckFlag (type:string)	No	MOBILE ORIGNATED (MO) ACKNOWLEDGMENT: This setting is used in conjunction with a Direct IP Delivery Destination. When true, the Gateway expects to receive an acknowledgement from the receiving server that it has received the message. If the Gateway does not receive an acknowledgement, it will attempt to re-deliver the message for 12 hours. No other messages will be delivered to the receiving server until the first message is acknowledged. TRUE – Enabled

BETA

Element/Field	Required?	Description
		FALSE - Disabled

14.8 MT Filters

Mobile Terminated (MT) filters provide white list capability for messages that can be received by the device. They allow a user to specify any combination of E-mail, Domain, or Direct IP sources which are permitted to send data to the destination IMEI.

By default, no white list options are provisioned and MT messages will be accepted from any source. To enable MT message source filtering, an entry must be added to the white list. Once an entry is specified for an IMEI, all MT payloads from sources not listed will be rejected. Up to 10 entries are supported per user, and a mix of rule types is permitted.

Wildcard characters (*) are permitted in the address entries. Wildcard characters may be placed either at the beginning or at the end of the address string, but not in the middle.

Some examples of valid email filters are shown below:

- E-mail Address: Tom@VendorDomain.com
- Wildcard: *Smith@VendorDomain.com, Tom@BackupDomain.*

Some examples of valid domain filters are shown below:

- Domain: VendorDomain.com
- Wildcard:*Domain.com

Some examples of valid direct IP addresses are shown below. Wildcards can be used only in the first or last quadrant of the IP address:

- IP Address: 168.45.34.4
- Wildcard: *.45.34.9, 168.45.34.*

When an MT payload is received from an unlisted source, a rejection is returned. For email message, the returned email indicates that the source failed to pass the particular user's filter. For Direct IP, a new error code (-10) is returned. The SBD Direct IP Specification has been updated accordingly (version 1.3).

Element/Field	Required?	Description
mtFilter (type:mtFilterImpl)	Yes	Mobile Terminated Filter
ruleType (type: mtFilterRuleTypeEnum)	Yes	See Filter Rule Type.
address (type:string)	Yes	Based on the ruleType specified, this should represent the IP, email, or domain you would like to white list.

BETA

Element/Field	Required?	Description
		See examples above for valid values.

14.9 Prepaid Voucher

This is the object that represents the prepaid voucher object. (prePaidVoucherImpl)

Element/Field	Required?	Description
voucherDenominationId (type: string)	Yes	Denomination associated with voucher type gathered from getVoucherDenominations
voucherActionType (type: voucherActiontypeEnum)	Yes	See VoucherAction Types.
voucherQuantity (type: string)	Yes	Quantity of vouchers to activate.
thirtyDayExtQty (type: string)	No	Quantity of 30 day extension vouchers to purchase.

14.10 Two Stage

This is the object that represents the two stage dialing object. (twoStageImpl)

Element/Field	Required?	Description
twoStageEnable (type: Boolean)	Yes	Boolean value that states if two stage dialing is enabled. TRUE – Turn on two stage dialing for this line. FALSE - Do not turn on two stage dialing for this line. Not allowed on CREW lines.
twoStagePin (type: string)	No	Four digit numeric pin used for two stage dialing

14.11 User

This is the object that represents the user account object for SPNet Pro access. (userImpl)

Element/Field	Required?	Description
userid (type: string)	Yes	User Id.
username (type: string)	Yes	Username
status (type: UserStatusEnum)	Yes	

BETA

Element/Field	Required?	Description
passwordExpiry (type: string)	No	Expiry date for password.
lastLogin(type: string)	No	Date of last user access
internalFlag (type: boolean)	Yes	Indicator if user is Iridium internal or not. TRUE: Iridium internal user FALSE: External user account
shadowAccess (type: Boolean)	Yes	Indicator if shadow access privileges are set for this user. TRUE: shadow access granted FALSE: shadow access privileges not allowed.
contactInfo (type: contactInfoImpl)	Yes	
firstName (type: String)	Yes	User first name.
lastName (type: string)	Yes	User last name
email (type: string)	Yes	Email address
contactPhone (type: string)	No	Phone Number
password (type: string)	Yes	
userAccessInfo	No	Array of access details
userSPAccess (type: userAccessInfoImpl)	No	
spAcct (type: string)	No	Service Provider account number.
serviceProviderName (type:string)	No	Service Provider Name
accessGroups (type: groupImpl)	No	Array of group access for the user/sp association
id (type: string)		If of access group
name (type: string)		Name of access group
owner (type: string)		Owner of group
pages (type: pageImpl)		Specific page reference. See Pages
status (type: string)		Status of access group
denialPages	No	Array of pages user has specific access denied, regardless of group privileges

DRA

Element/Field	Required?	Description
denialPage (type: pageImpl)		Specific page reference. See Pages
spPoid (type: string)	No	SP reference ID
preferredAccount (type: string)	No	If user has access to multiple SP accounts, this is the default login account
preferredLandingPage (type: string)	No	If set this will be the default page after login access is granted for SPNet
admin (type: Boolean)	Yes	TRUE: user has admin access FALSE: user does not have admin access

14.12 Page

This is the object that represents the SPNet Page object. (pageImpl)

Element/Field	Required?	Description
id (type: string)	Yes	Page ID
name (type: string)	Yes	Page Name
parentPageId (type: string)	Yes	Parent Page within SPNet Menu
pageLevel (type: string)	No	Page Level 1: Tab 2: Header 3: Page 4: Action
permissionReqFlag (type: string)	Yes	Indicates if specific permission is required to access this page. TRUE: permission required FALSE: no special permission required to access this page, will be available to all users.
status (type: operationStatusEnum)	Yes	Status of the page
menuOrder (type: string)	Yes	Order within SPNet Menu
visible (type: string)	Yes	Visible within SPNet
module (type: string)	Yes	SPNet module reference
controller (type: string)	No	SPNet controller reference
action (type: string)	No	Action within page
applicableActions (type: string)	No	Available actions for the page

BETA

Element/Field	Required?	Description
breadcrumb (type: string)	No	Breadcrumb display string
serviceType (type: string)	No	Service type
resourceid (type: string)	No	Resource reference

BETA