

Access-i Developer Guide

Version 1.1.2 for NX

Last Change: 28.06.2021

R. Schneider, D. Franger
SHS DI MR AST

11337213-ASD-ACI-01

Copyright by Siemens Healthcare GmbH

Table of Contents

| | |
|---|----------|
| 1 Introduction..... | 5 |
| 1.1 Purpose of Document | 5 |
| 1.2 Area of validity of the document | 5 |
| 1.3 History of Changes | 5 |
| 1.4 Definitions and Abbreviations..... | 5 |
| 1.4.1 Definitions | 5 |
| 1.4.2 Abbreviations..... | 5 |
| 1.5 References | 6 |
| 2 General description | 7 |
| 2.1 Intended Use of Product | 7 |
| 2.2 Access-i Interface Overview..... | 7 |
| 2.3 Communication..... | 8 |
| 2.3.1 HTTPS Communication | 8 |
| 2.3.2 Secure WebSocket Communication | 11 |
| 2.4 Setup of Access-i Protocols | 14 |
| 2.4.1 Enable Interactivity for a Specific Template | 15 |
| 2.5 Setup of Access-i Programs..... | 16 |
| 2.6 Client Implementation Guideline | 17 |
| 2.6.1 Checking for Access-i Server Availability | 17 |
| 2.6.2 Registering with Access-i..... | 18 |
| 2.6.3 Establishing WebSocket Connection and connecting relevant Services to WebSocket..... | 19 |
| 2.6.4 Obtaining Host Control | 20 |
| 2.6.5 Retrieving General Information..... | 21 |
| 2.6.6 Retrieving Existing Templates and Loading New Templates into Queue | 22 |
| 2.6.7 Modifying and Starting Templates | 23 |
| 2.6.8 Handling Issues..... | 24 |
| 2.6.9 Receiving Images and Tracking Data | 25 |
| 2.7 Access-i Services | 26 |
| 2.7.1 Remote Service..... | 26 |
| 2.7.2 Authorization Service..... | 28 |
| 2.7.3 HostControl Service..... | 31 |
| 2.7.4 Patient Service | 39 |
| 2.7.5 SystemInformation Service..... | 45 |
| 2.7.6 TemplateExecution Service | 55 |
| 2.7.7 TemplateSelection Service | 64 |

| | |
|--|------------|
| 2.7.8 TemplateModification Service..... | 74 |
| 2.7.9 Parameter/standard Service | 82 |
| 2.7.10 Table Service | 164 |
| 2.7.11 Image Service | 173 |
| 2.7.12 Tracking Service..... | 191 |
| 2.7.13 Issue Service..... | 199 |
| 2.7.14 Adjustment Service..... | 209 |
| 2.7.15 Parameter/configured Service | 220 |
| 2.7.16 Debugging Service | 233 |
| 2.7.17 Time Service | 238 |
| 3 Software Development Kit | 239 |
| 3.1 Installation | 239 |
| 3.2 Access-i Simulator | 240 |
| 3.3 Access-i Test Client | 242 |
| 3.3.1 Remote Service..... | 243 |
| 3.3.2 Authorization Service..... | 243 |
| 3.3.3 Host Control Service..... | 244 |
| 3.3.4 Template Selection..... | 244 |
| 3.3.5 Template Modification | 245 |
| 3.3.6 Template Execution..... | 247 |
| 3.3.7 Standard Parameter Service | 248 |
| 3.3.8 Table Service | 249 |
| 3.3.9 Image Service | 250 |
| 3.3.10 Tracking Service..... | 251 |
| 3.3.11 Patient Service | 251 |
| 3.3.12 Issue Service..... | 252 |
| 3.3.13 Configured Parameter | 253 |
| 3.3.14 System Information..... | 253 |
| 3.3.15 Time Service | 254 |
| 3.3.16 Log Window..... | 254 |
| 3.3.17 Smart HTTP | 255 |
| 4 Annex..... | 256 |
| 4.1 Recommended Libraries for Remote Clients..... | 256 |
| 4.2 Configuration of the Configurable Parameter Card | 257 |
| 4.3 OpenSource Licenses..... | 261 |
| 4.3.1 Websocket-sharp | 261 |
| 4.3.2 Fleck | 261 |
| 4.3.3 Fellow Oak DICOM..... | 262 |

| | |
|---------------------|-----|
| 4.4 Changelog | 270 |
|---------------------|-----|

1 Introduction

1.1 Purpose of Document

This document is intended to provide a comprehensive documentation of the Access-i MR Scanner Interface to a 3rd party developer who wants to integrate against this interface.

1.2 Area of validity of the document

The requirements are valid for following system types:

- MAGNETOM Sola
- MAGNETOM Sola^{Fit}
- MAGNETOM Vida
- MAGNETOM Vida^{Fit}
- MAGNETOM Altea
- MAGNETOM Lumina

1.3 History of Changes

| Version Date | Author, Dept. | Reason |
|-----------------|--------------------|-------------------------------|
| V01, 2019-03-19 | Dirk Franger (ext) | Initial version for Numaris X |

1.4 Definitions and Abbreviations

1.4.1 Definitions

Technologist Primary user of the MR system

1.4.2 Abbreviations

| | |
|------------|---|
| RC | Remote Client |
| LAN | Local Area Network |
| REST | Representational State Transfer |
| JSON | JavaScript Object Notation |
| IRT | Interactive Realtime |
| Fast Track | An information pathway for providing MR data without utilizing the MR |

| | |
|------------------|---|
| | Host image DB. |
| Template | A “template” MR protocol connected to an MR sequence, an instantiation of which is intended for remote execution. |
| Template Live-Id | A Template is used in two different contexts: Offline within a program as configured in the DotCockpit and online as a running instance in the program queue. Depending on the context a template may have two different types of ids: Offline in the DotCockpit a configured template has a static template id whereas online in the queue it has a dynamic “live-id” that changes each time the template is loaded into queue either manually or via <code>templateSelection/loadTemplateSteps</code> (also see section 2.7.7.2.10). |

1.5 References

- [1] 11104856-ASD-3PN-01, 3rd Party Network Connection
- [2] 11104856-ASD-HPS-01, In-plane „Half Pixel Shift“ due to Fourier Transformation
- [3] 11104856-ASD-POP-01, List of Issues supported by Access-i
- [4] 11104856-ASD-ACI-01, Access-I developer guide version 1 for N4 VE11C-AP03

2 General description

2.1 Intended Use of Product

The purpose of Access-i is to allow 3rd party clients to establish a bi-directional communication with the MR scanner. The physical connection to the MR scanner is established via a dedicated LAN router – for further details please refer to Ref. [1].

With Access-i 3rd party clients are enabled to remotely control the MR scanner in the sense of parameterizing, executing and retrieving image and projection data from arbitrary pulse sequences. Moreover, patient and system data can be retrieved in order to establish a complete workflow and business logic beyond the MR host and Exam task.

2.2 Access-i Interface Overview

Access-i contains a number of dedicated remote service interfaces that allow a RC application to control parameters and examinations on the MR scanner.

These interfaces are subdivided according to their purpose:

- Adjustment Service
- Authorization Service
- Debugging Service
- Host Control Service
- Image Service
- Issue Service
- Parameter /configured Service
- Parameter /standard Service
- Patient Service
- Remote Service
- Table Service
- Template Execution Service
- Template Modification Service
- Template Selection Service
- Time Service
- Tracking Service
- System Information Service

The communication between the remote client and the server on the scanner is REST- and JSON-based, i.e. the client issues each command via an HTTP request and receives an HTTP response that contains the answer to the request in JSON format.

2.3 Communication

2.3.1 HTTPS Communication

Access-i provides its operations via a secure HTTPS-based REST interface that supports HTTPS GET and HTTPS POST requests, i.e. the RC issues each command via an HTTPS request and receives an HTTPS response that contains the answer to the request in JSON format.

According to general REST guidelines, operations that just retrieve information from the host are called using HTTPS GET requests. Operations that change the state on scanner, on the other hand, are called using an HTTPS POST request.

The Access-i server is listening on **port 7787**.

SSL Certificate

Access-i uses a private SSL certificate which has not been issued by an official certificate authority. In your HTTP client implementation this may lead to an event / callback that says that the certificate is not trustworthy. You need to confirm this event / callback by returning "true", thus saying that you trust this certificate. In C# this looks as follows when the `HttpWebRequest` class is used:

```
request.ServerCertificateValidationCallback =
    (sender, certificate, chain, sslPolicyErrors) => true;
```

TLS Encryption Algorithm

In software versions **VE11C-AP03** and **VE11E TLS1.0** is used to encrypt HTTPS communication between the Access-i server and the remote client. Starting with VE11E SP1 and in VA20A **TLS1.2** is used.

On the remote client side you can enforce this by explicitly setting the SSL encryption algorithm within your application to TLS1.2. E.g. in C#, this is done as follows:

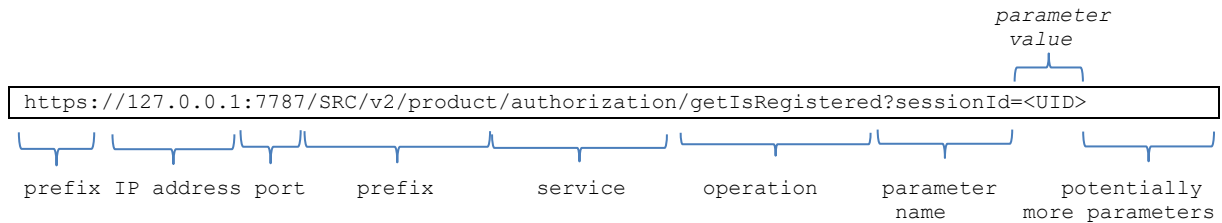
```
System.Net.ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls12;
```

HTTPS-GET requests

For HTTPS-GET requests, the URL of the HTTPS call contains all information needed to execute the operation, which is:

- https prefix
- server IP address
- server port
- prefix:
 - o the "SRC" tag
 - o SRC version number ("v2")
 - o the "product" tag
- service name (name of the service the operation belongs to)
- operation name
- optional parameters provided to the operation

Example:



In this example, the server has the IP address 127.0.0.1 and listens on port 7787.

The HTTPS request calls the `getIsRegistered` operation of the `authorization` service and provides a `sessionId` as parameter.

The format of parameters in HTTPS-GET requests are URL compliant, i.e.

- operation name and the name of the first parameter are separated by a question mark (?)
- operation name and operation value are separated by the equal sign (=)
- name/value parameter pairs are separated by an ampersand (&)

Be aware that parameter values that are strings are not surrounded by quotes.

HTTPS-POST requests

HTTPS-POST requests also contain server IP address, port, service name and operation in their URL, but the operation's parameters are transferred in the message body of the HTTPS-POST request, encoded in JSON format.

To indicate to the server that parameter values are provided in the message body in JSON format, the content-type of the HTTPS-request must be set accordingly:

```
Content-Type: application/json
```

Example:

To register with the SRC server, the RC must call the `register` operation of the `Authorization` service. To do this, the client must provide a valid license which is passed as parameter to the operation:

```
https://<SERVER_IP>:7787/SRC/v2/product/authorization/register
```

Content-Type

```
Content-Type: application/json
```

Message Body (in JSON format)

```
{"license": "<AUTHENTICATION_STRING>"}
```

HTTPS Responses

Each HTTPS request sent by the client is followed by a HTTPS response returned from the Access-i server to the client. This response is also encoded in JSON format and always (at least) indicates whether the operation was successful:

```
{"success":true}
```

The response may also contain multiple return values that are represented as key/value pairs, e.g.:

```
{"result":{"success":true,"reason":"ok","time":"20171212T112748.916"},  
"sessionId":"60ad085e-54d3-4cf7-a5cb-e5d07eaf6f63","privilegeLevel":"Advanced"}
```

2.3.2 Secure WebSocket Communication

Similar to TCP connections, WebSockets allow sending data bidirectionally between two endpoints. Furthermore, string-based communication is possible, so sending JSON-formatted messages between server and client can be accomplished easily.

Regular polling of service methods consumes unnecessary resources both on server and client side and makes the client development more complicated and less elegant.

A far better way for the client is to just sit and wait until a message arrives that signals a state update on the server. In Access-i this is done via secure WebSockets which are a separate connection between the server and client and which are exclusively used by the server to send state update messages to each client that uses WebSockets.

Messages that are sent from the server to the RC nearly have the same format as the corresponding responses of HTTPS requests, in other words: The client gets a "response" via the WebSocket channel without sending the corresponding request.

The WebSocket on the SRC server side listens and sends on **port 7788**. Before a RC may connect to this WebSocket it needs to successfully register via the Authorization service. The returned session id then is required to successfully establish a WebSocket connection to the SRC server.

TLS Encryption Algorithm

The WebSocket connection is encrypted using TLS 1.2 encryption. This was newly introduced in Access-i version 2 and in Access-i version 1 on MR software version **VE11E SP1**. In the first editions of Access-i version 1 TLS 1.0 (which is regarded as being unsafe) encryption was used. In other words, **WebSocket and HTTPS connections always use the same TLS encryption algorithm, dependent on the MR Host software version.**

If your software needs to be able to connect to multiple Access-i versions, your client should offer TLS1.0, TLS1.1 and TLS1.2 as supported encryption algorithms and let the server decide which TLS encryption to use. This is typically done automatically during the connection setup handshake. Please see the green code line in the next section as example.

WebSocket Connection Setup

The following paragraph describes the minimal steps needed to establish a WebSocket connection to the SRC server. Please look for a WebSocket library for your programming language of choice. In the following example the WebSocketSharp.NET library is used for the C# programming language.

1. Register via the Authorization service / Register HTTPS-POST request:

```
https://127.0.0.1:7787/SRC/v2/product/authorization/register
```

Request body:

```
{ "license": {"Name":"Test Authorization
Key", "Comment":null, "WarnDate":"20180601", "StartDate":"20171109", "ExpireDate"
:"20180630", "SystemId":null, "IsReadOptionAvailable":true, "IsExecuteOptionAvai
lable":true, "IsAdvancedOptionAvailable":true, "Version":"1.0", "Hash":"VX7l%2BV
isdUjbfz40S5f%2FRCTKMM453ZGeQxi8jvlPdYrNWyDStu9oKVtrs%2Bv9iT9Ys4etegseQMDymkZ
JcaXP%2BA%3D%3D"}, "name": "Test Remote Client" }
```

- Remember the session id which is included in the response of the Register request:

```
{ "result": { "success": true, "reason": "ok", "time": "20171212T112748.916" },
  "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763",
  "privilegeLevel": "advanced" }
```

- Open the WebSocket and connect to the SRC server (the SRC WebSocket listens on port 7788 and on the URL subnode „SRC“), providing the session id:

wss://127.0.0.1:7788/SRC?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763

In the following C# code snippet a WebSocket is established:

```
private WebSocketSharp.WebSocket clientWebSocket;

private void Connect()
{
    string url = "wss://127.0.0.1:7788/SRC?sessionId=" +
                "26eb060c-553d-4c15-bf5b-23cf29012763";
    clientWebSocket = new WebSocketSharp.WebSocket(url);
    clientWebSocket.SslConfiguration.EnabledSslProtocols =
        SslProtocols.Tls | SslProtocols.Tls11 | SslProtocols.Tls12;
    clientWebSocket.OnMessage += OnClientWebSocketMessage;

    clientWebSocket.Connect();
}

private void OnClientWebSocketMessage(object sender, MessageEventArgs e)
{
    try
    {
        dynamic message = JObject.Parse(e.Data);
        string service = message.service;
        string request = message.request;
        string response = message.response.ToString();
        Console.WriteLine("Service: " + service);
        Console.WriteLine("Request: " + request);
        Console.WriteLine("Response: " + response);
    }
    catch (Exception ex)
    {
        Log.Error("Parsing WebSocket message failed.", ex);
    }
}
```

- By default, WebSocket messages are deactivated for all services (because a specific RC may not want to receive messages from all services), i.e. you have to explicitly enable WebSocket messages for each service that is supposed to send those to your client application. This is done via the connectServiceToDefaultWebSocket service method that is available for all services that support WebSockets, e.g.:

https://127.0.0.1:7787/SRC/v2/product/patient/
connectServiceToDefaultWebSocket?sessionId=26eb060c-553d-4c15-bf5b-
23cf29012763

Using Multiple WebSockets per RC

It is possible to use more than one WebSocket per RC. This can be useful for routing messages of different services to different WebSockets that handle different tasks of your application.

To enable this feature you need to do 2 things:

1. Provide an id for the WebSocket when the WebSocket connection is established (at least for the 2nd WebSocket that is established):

```
wss://127.0.0.1:7788/SRC?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=My2ndWebSocket
```

C# example:

```
string url = "wss://127.0.0.1:7788/SRC?sessionId=" +  
            "26eb060c-553d-4c15-bf5b-23cf29012763&id=My2ndWebSocket";  
clientWebSocket = new WebSocketSharp.WebSocket(url);  
// ...
```

2. Connect the service that should be routed to this second WebSocket by calling the ConnectServiceToWebSocket message of this service which takes the WebSocket's name as parameter, e.g.:

```
https://127.0.0.1:7787/SRC/v2/product/patient/  
connectServiceToWebSocket?sessionId=26eb060c-553d-4c15-bf5b-  
23cf29012763&id=My2ndWebSocket
```

The WebSocket that has been established without a specific id is the default WebSocket to which all service WebSocket messages are routed to if not specified otherwise. It is recommended to establish a default WebSocket first before establishing another (named) one.

WebSocket Connection Teardown

Before closing the websocket connection you should first deregister from Access-I via authorization/deregister. Closing the websocket from the remote client side is interpreted by the Access-i server that the client died and therefore the client is automatically deregistered.

2.4 Setup of Access-i Protocols

Arbitrary diagnostic sequences can be made available to the RC by attaching the Access-i Dot-AddIn within the Dot Cockpit (Figure 1). Remote parameterization and images are automatically available for the RC if the step is put in the queue.

NOTE: Service sequences and imaging sequences which require another Dot-AddIn are not supported by Access-i.

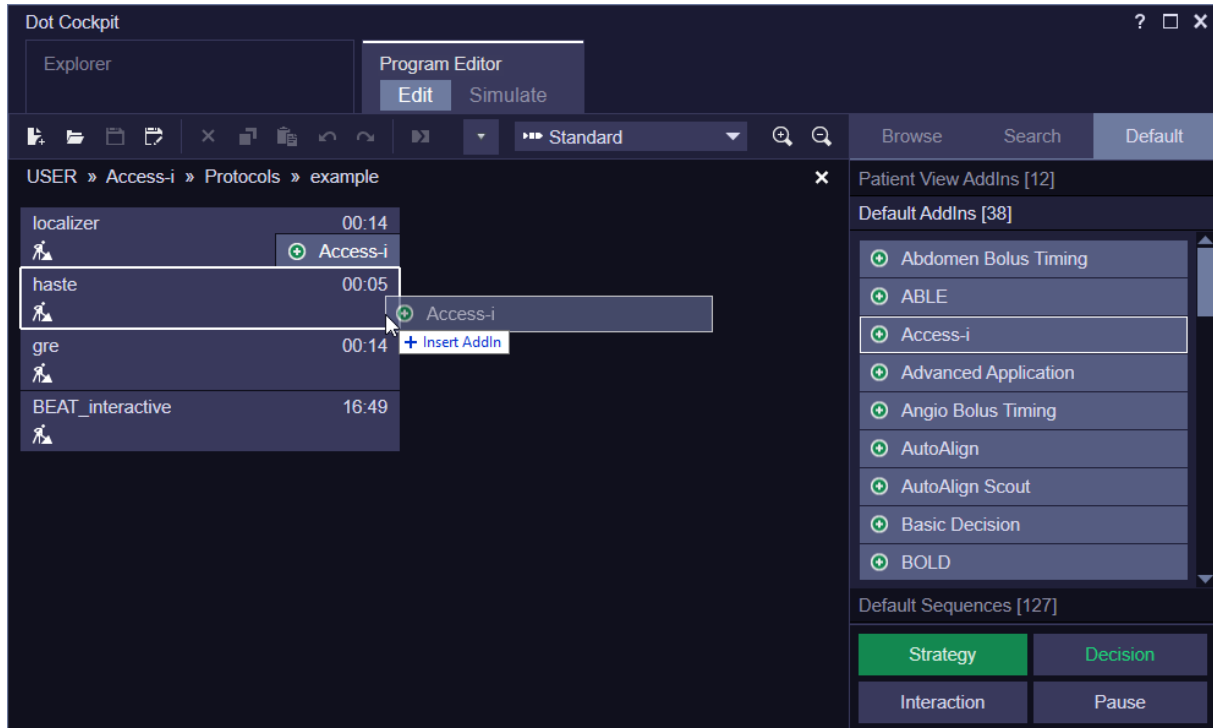


Figure 1: Configuring imaging sequences for remote control.

Access-i further supports the remote execution of interactive realtime imaging sequences, such as the 'BEAT_interactive' sequence. If the user wants to use the IRT imaging sequence in interactive mode, the property has to be explicitly activated in the Dot-AddIn Configuration.

2.4.1 Enable Interactivity for a Specific Template

1. Double-click the corresponding interactive step (BEAT_interactive in this example). In the following step properties window, click the “AddIn Configuration...” button:

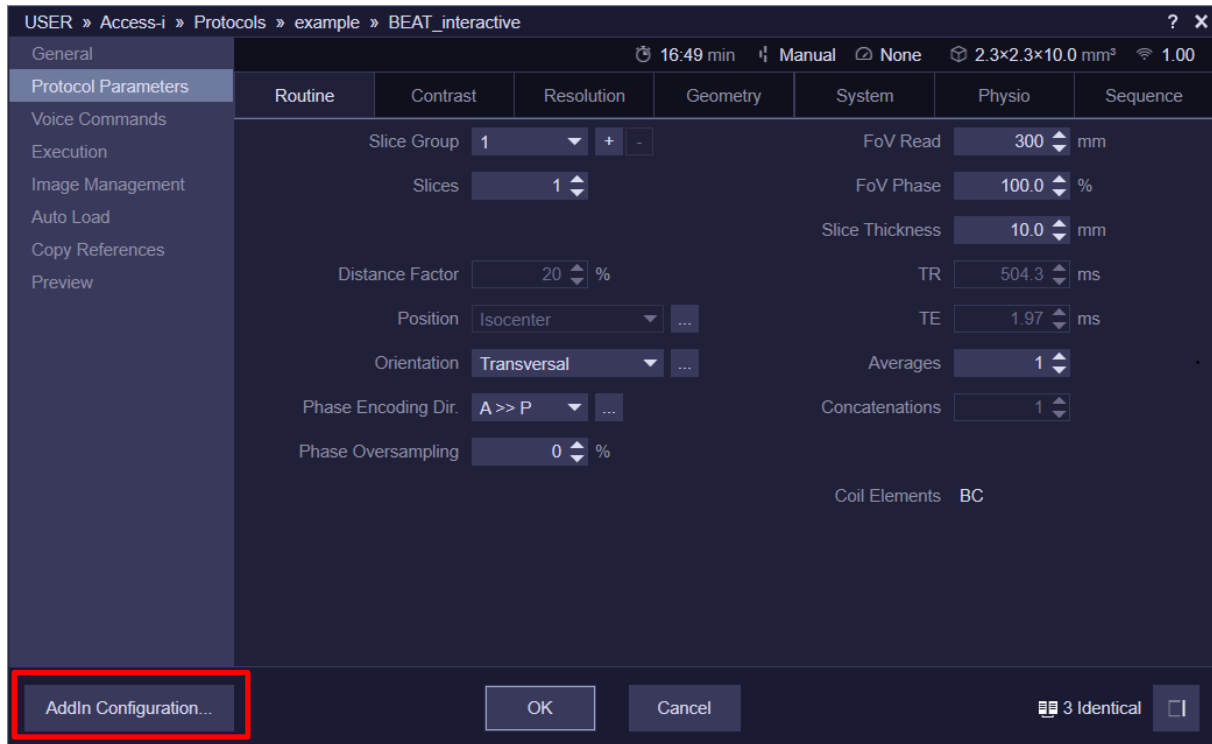


Figure 2: Configuration of Access-i AddIn to run an IRT imaging sequence in an interactive fashion. After double-clicking the desired step in the Dot Configurator you need to click “AddIn Configuration...” in the lower left corner of the step properties dialog.

2. In the following Dot AddIn Configurator window, check “Enable Interactivity” on the Access-i configuration tab:

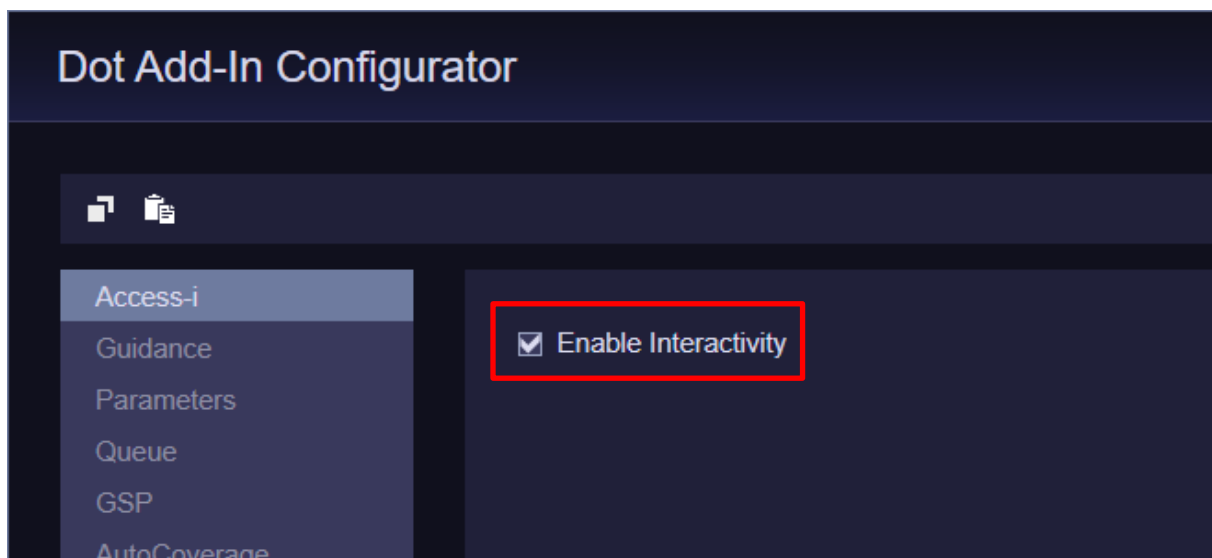


Figure 3: Checking “Enable Interactivity” in the Access-i configuration tab

2.5 Setup of Access-i Programs

To generally enable a measurement program for remote access, you need to adjust the directory properties.

- Open the Dot Cockpit > Explorer task > Browse subtask and select the measurement program (directory) in the tree view.
- Right-click and choose Edit Properties from the context menu. The Directory Properties dialog box opens.
- In the Description field of the Directory Properties dialog box - enter: [Remote]
- Click OK to confirm

Directory Properties

USER » Access-i » Protocols » example

Name: example

Description: [Remote]

Body Part Examined: - [Manage](#)

Table Positioning Mode: <derive>

Archive: ☒

Visibility: ☒ Show irrespective of licensing

OK Cancel

Figure 4: Configuring Access-i measurement programs.

The measurement program is prepared for remote access. Now you can remotely add this program to the measurement queue. To control the program steps, you must additionally apply the Access-i Dot add-in.

2.6 Client Implementation Guideline

The client implementation guideline is supposed to support the remote client developer in understanding what a typical client application needs to do to smoothly integrate with the Access-i interface. It is based on experiences that we gathered during our prototyping and implementation phase together with first collaboration partners and some prototype clients and should help you to better understand how the architecture of an Access-i client may look like and help to get up and running.

The services in this documentation are ordered according their relevance and the timely order the client needs them to perform a whole workflow.

In the figures used in this section HTTPS calls are shown in blue (e.g. `remote/getIsActive`) whereas HTTPS responses received by the RC are colored green (e.g. `"success":true`).

2.6.1 Checking for Access-i Server Availability

From a client-application perspective the first thing you want to do is to check whether the Access-i server is running and reachable. This is normally done using the 'getIsActive' method of the Remote service (see chapter 2.7.1). As long as this http-call either times out or returns false the client app can indicate to its user that the Access-i interface is not reachable.

In Figure 5 you can see that in the beginning the server is not reachable (Access-i not running yet, network cable not connected, client connections not allowed,...), so client HTTPS-calls to the `remote/getIsActive` time out. As soon as a response is received by the RC it knows that the Access-i server is reachable and active, so it may proceed registering using the Authorization service (see chapter 2.7.2).

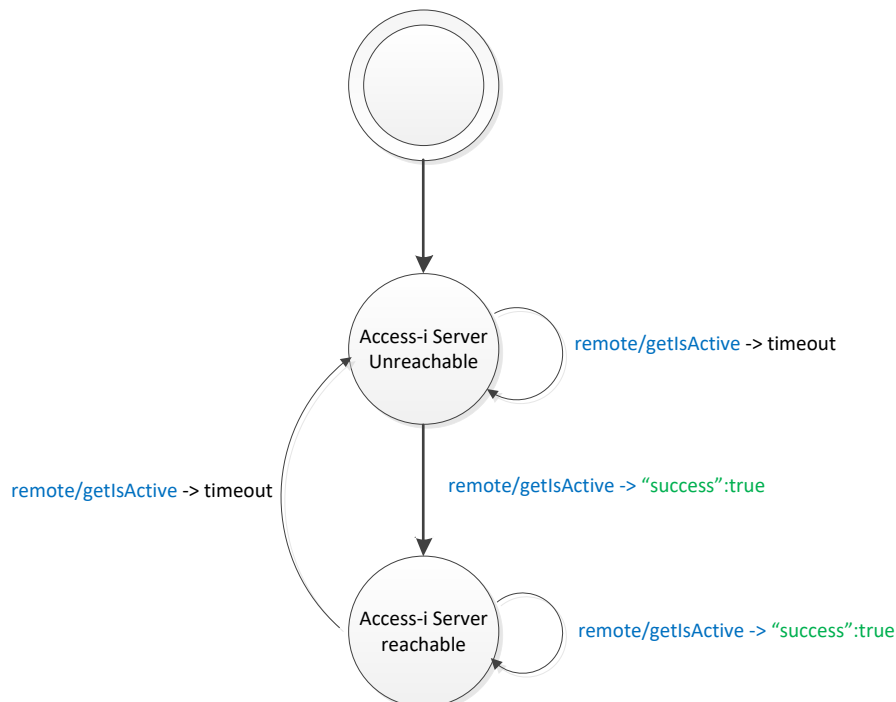


Figure 5: Polling whether server is active and reachable

Access-i for Sola/Vida (Numaris X, version “2.0”) is the second Access-i version after *Access-i for Aera/Skyra/Prisma* (VE11 software line, version “1.0”). As the URL of all HTTPS requests in the new version contain “v2” (opposed to “v1” in the first version) and the interfaces slightly differ, it makes sense to ask the scanner about the actual Access-i interface version by calling `remote/getVersion` so that the RC software can adapt accordingly.

2.6.2 Registering with Access-i

Once the Access-i server is reachable, the RC may register using a valid authentication key via the Authorization Service (see chapter 2.7.2):

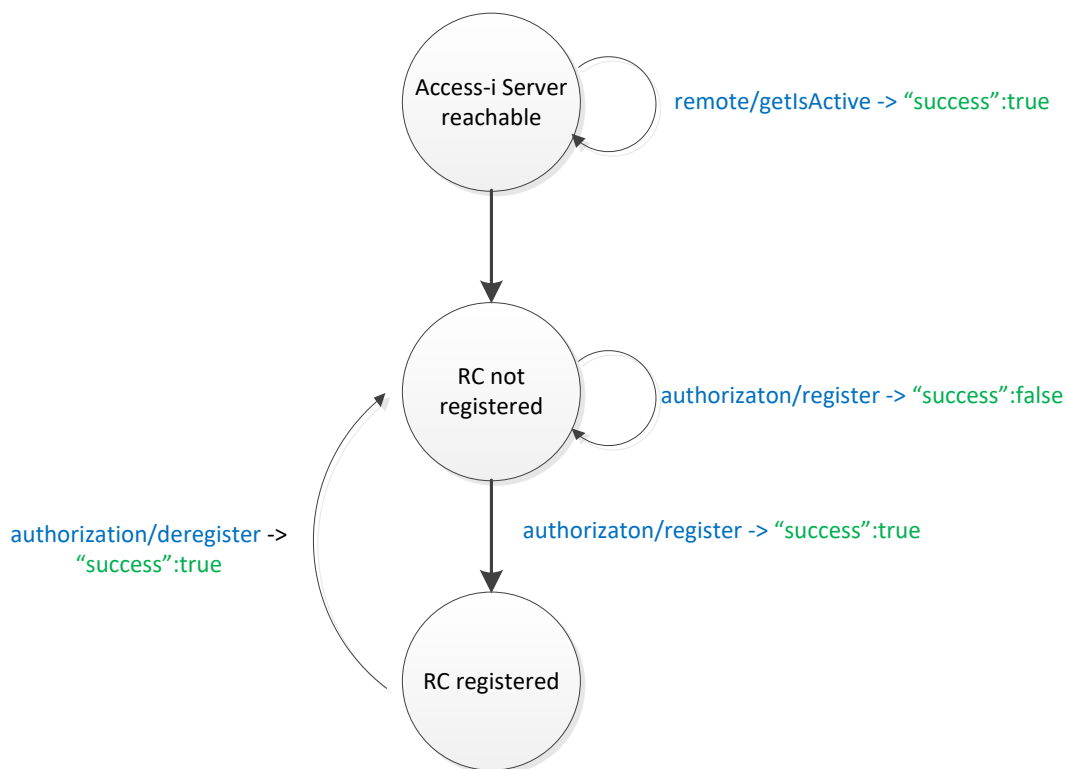


Figure 6: Registering with Access-i

2.6.3 Establishing WebSocket Connection and connecting relevant Services to WebSocket

Once the RC has successfully registered with the Access-i server it may establish a websocket connection to it which consists of 2 parts:

- Establishing a websocket connection as described in chapter 2.3.2
- Connecting each service relevant for the RC application to WebSocket connection
- If the RC closes the WebSocket connection, it is automatically deregistered by the Access-i server as the server thinks that the RC has died.

Therefore if you want to disconnect from Access-i, you should first deregister and then close the WebSocket connection.

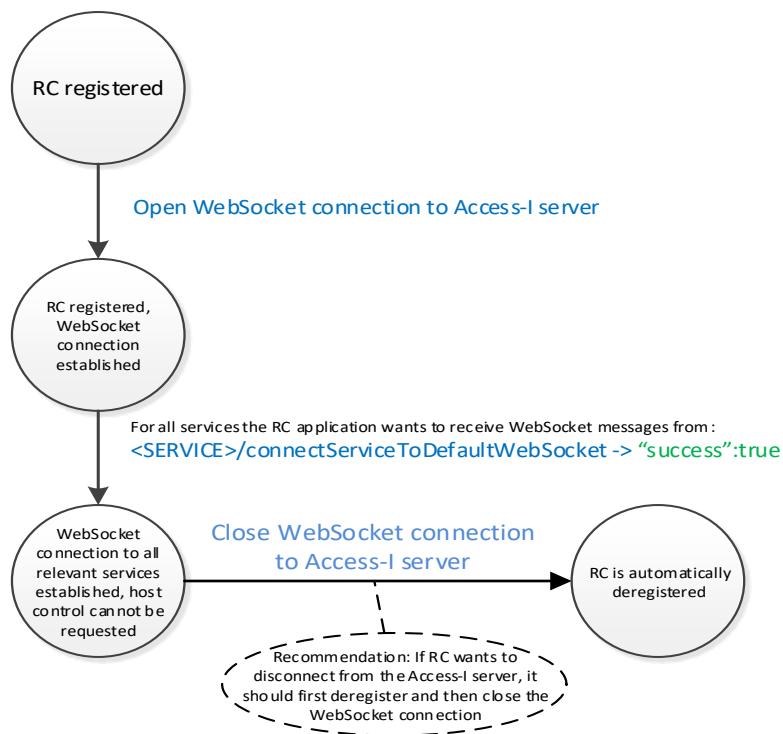


Figure 7: Establishing & Closing WebSocket Connection

2.6.4 Obtaining Host Control

After the RC has successfully registered and the WebSocket connection is set up in general it wants to execute some operations on the scanner. To do this it has to request host control (see chapter 2.7.3). This is done in 2 steps:

- Polling `hostControl/getState` (see 2.7.3.2.1) of the Host Control Service to find out whether host control can be requested (and getting the reason why it cannot be requested if it is not possible). Alternatively (and preferred approach), the RC waits for the corresponding `hostControl/getState` WebSocket message to change.
- Requesting hostControl by sending `hostControl/requestControl` (see 2.7.3.2.2)

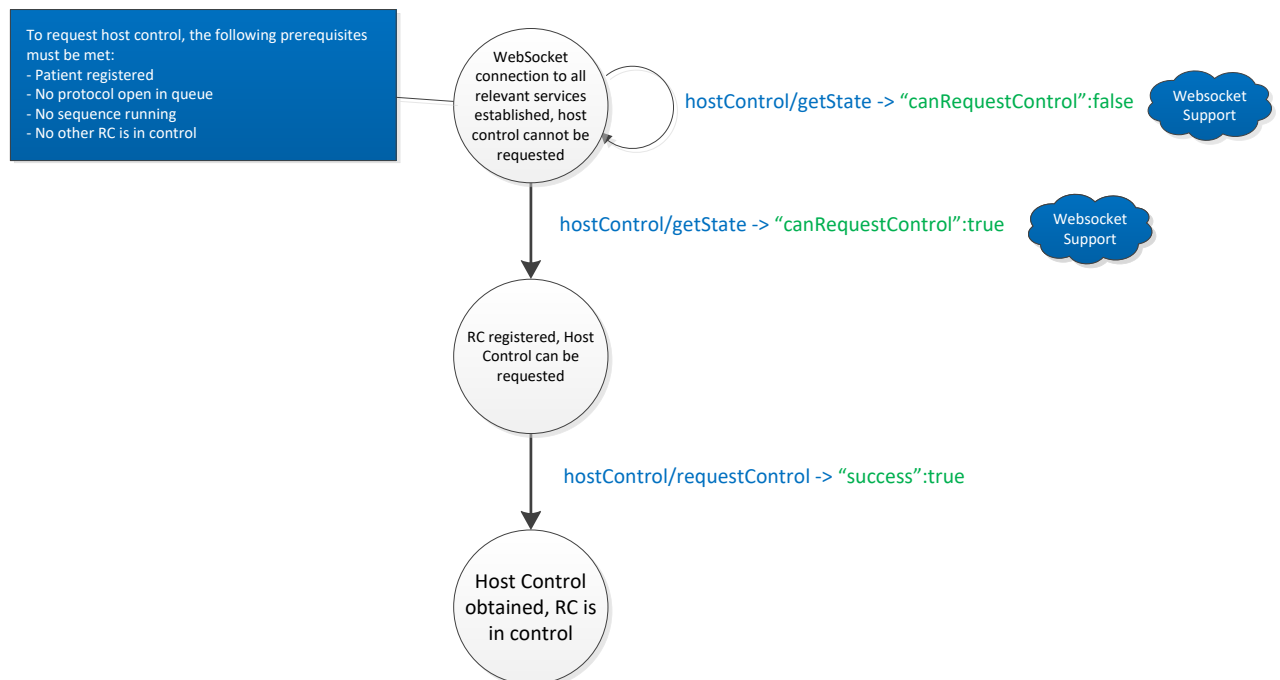


Figure 8: Requesting Host Control

2.6.5 Retrieving General Information

Now it is a good time for the RC to retrieve some general information like patient or system-specific information. To do that host control actually is not required. This is done using the Patient Service (see chapter 2.7.4) and System Information Service (see chapter 2.7.5).



Figure 9: Retrieving General Information

2.6.6 Retrieving Existing Templates and Loading New Templates into Queue

After the RC has registered and obtained control there may already be several templates loaded in the queue (because it was done during a previous control period or because they were loaded into the queue manually by the user directly at the MR Host).

So first thing the RC should do is to analyze which templates are already in the queue (templateExecution/getTemplates, see chapter 2.7.6.2.1). If the desired templates are not (yet) in the queue, the RC can load them in 2 steps:

1. Finding out which programs & steps can be remotely loaded into the queue “from the Dot Cockpit” using templateSelection/getTemplatePrograms and templateSelection/getTemplateSteps (see chapters 2.7.7.2.6)
2. Loading a whole program or one/multiple Template Steps into the queue using templateSelection/loadTemplateProgram or templateSelection/loadTemplateSteps (see chapters 2.7.7.2.9 and 2.7.7.2.10)

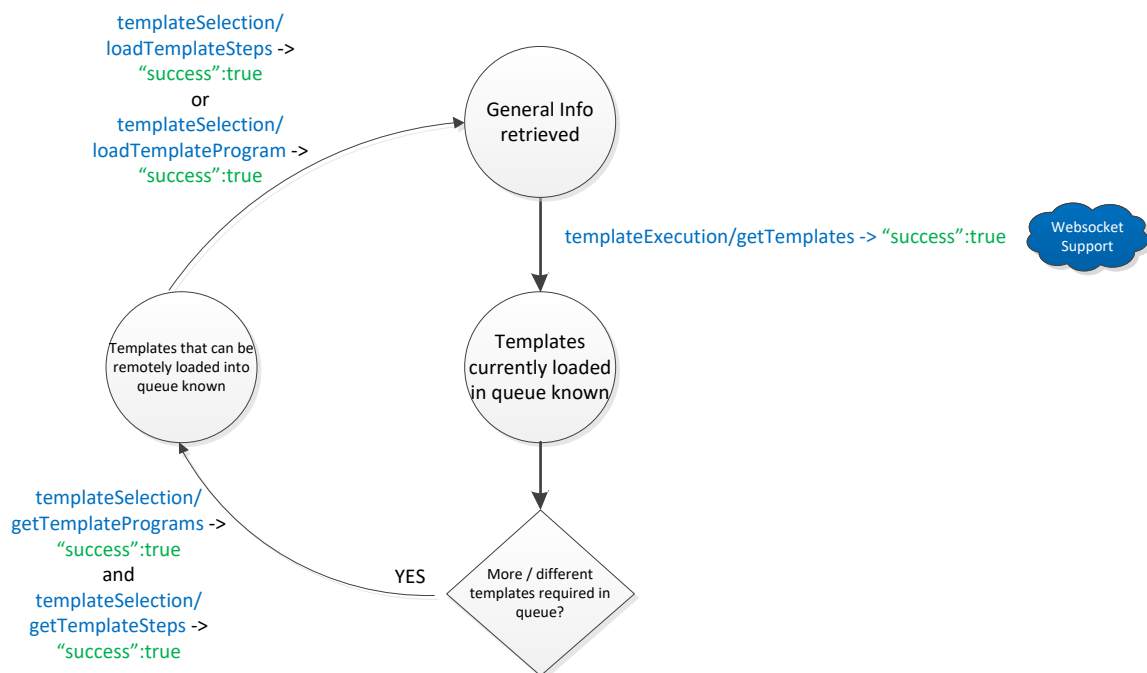


Figure 10: Retrieving Template State in Queue and Loading Templates into Queue

2.6.7 Modifying and Starting Templates

As the required templates are now in the queue (meaning they are active) they can be modified and/or started. Very often the flow is that a template is opened (see chapter 2.7.8.2.2) and some protocol parameters (like slice group position and orientation) are modified using the standard parameter service (see chapter 2.7.9). After all protocol modifications are finalized the template can be started using templateExecution/start (see chapter 2.7.6.2.4).

Please note that it is a prerequisite of a template to be open before any of its parameters can be modified using the parameter/standard or parameter/configured services. This also means that a template can be directly started using templateExecution/start if there is no need to modify it.

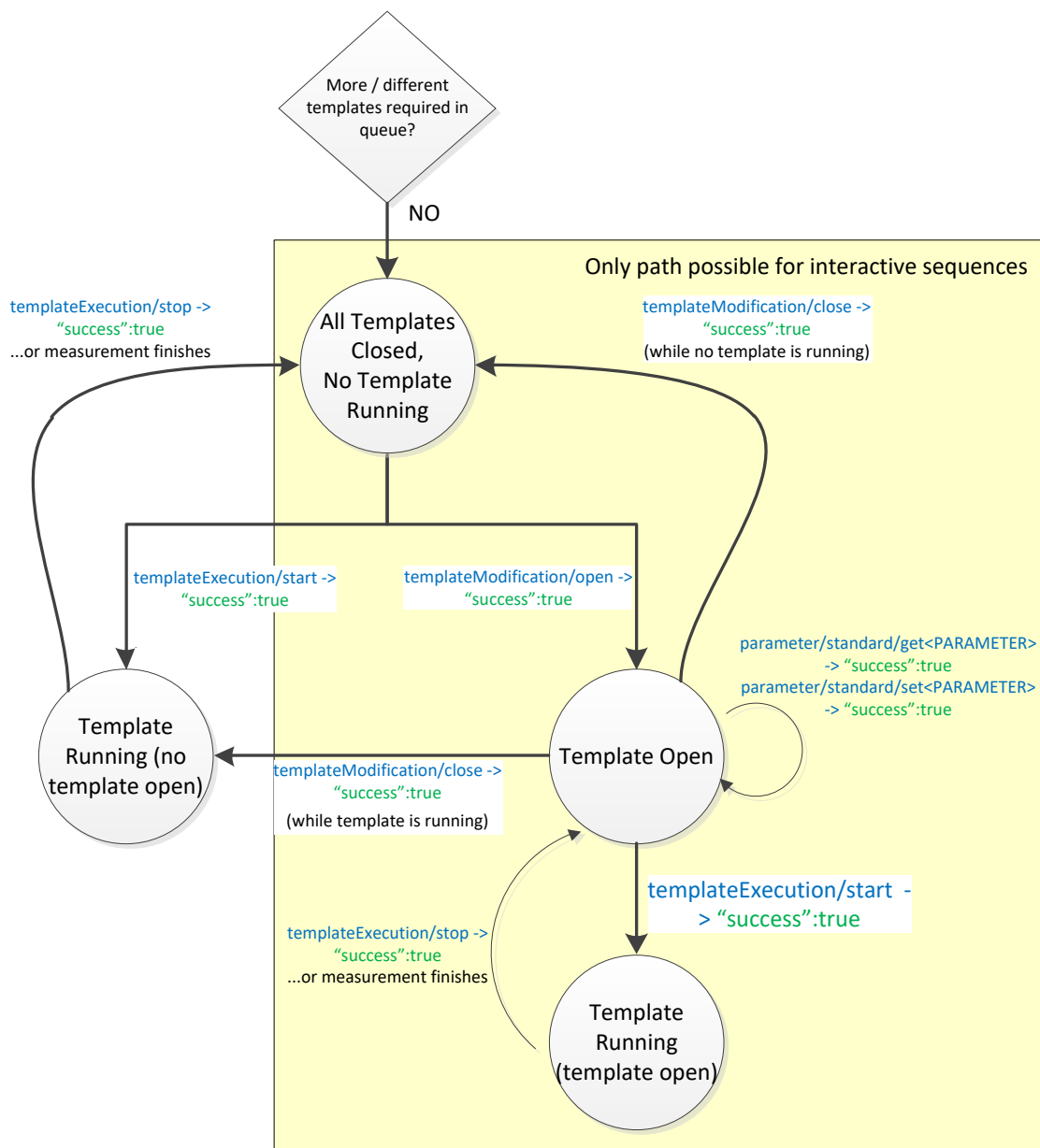


Figure 11: Modifying and Starting Templates

Hint:

Templates associated with interactive sequences need to open before they can be started; i.e. only the right path in Figure 10 (marked yellow) is possible.

2.6.8 Handling Issues

During the prepare phase of an MR measurement, i.e. when a template starts running and the RC is in control, issues may occur. Issues are specific incidents on the MR scanner that need approval by the user. If a RC is in control it is also responsible for either confirming or declining an issue. For more details regarding issues see chapter 2.7.13 .

Issues may also occur when adjustments are started (see Adjustment Service in chapter 2.7.14).

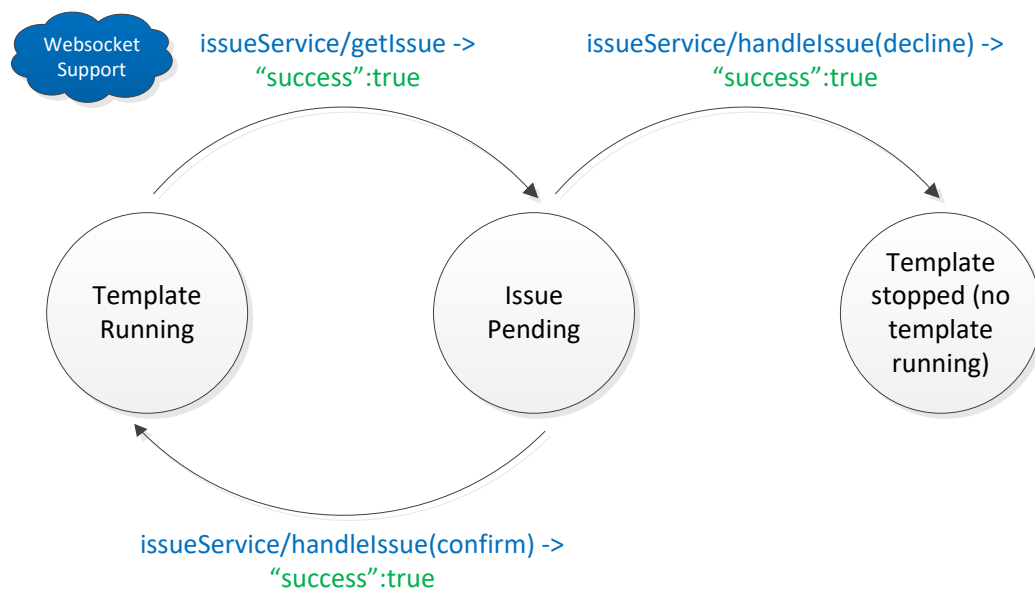


Figure 12: Handling Issues

2.6.9 Receiving Images and Tracking Data

As a consequence of running a template the RC may receive image and/or tracking data via WebSocket connection only. Depending on the type of sequence and the configuration of the template's protocol images or tracking data may be received while the template is still running and/or after the MR scan is finished.

In other words, running templates is the reason for a RC to receive scanned data, but the time when this happens is not strictly connected to start or finish of the scan, i.e. on the RC side the state machine / component responsible for receiving this data should not rely on points in time like measurement start or measurement finish.

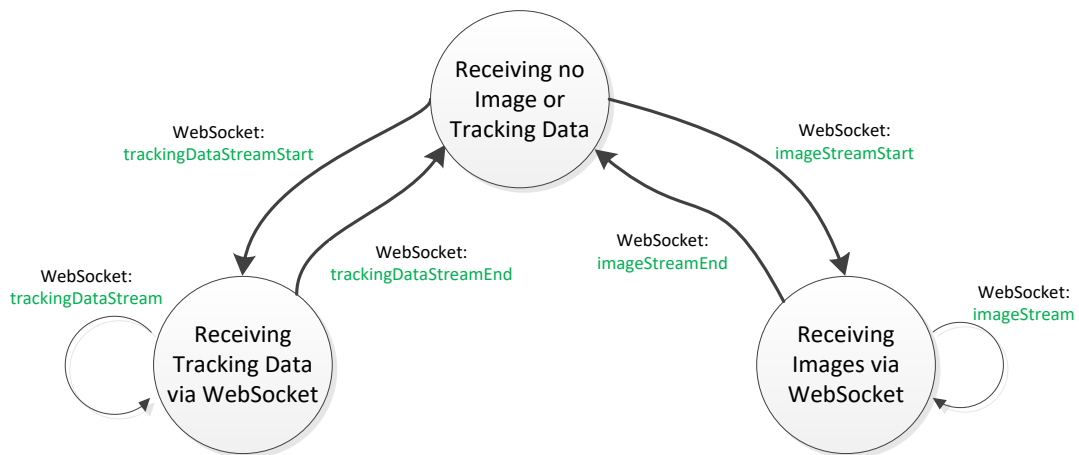


Figure 13: Receiving Image and Tracking Data

For further details regarding receiving tracking data please see chapter 2.7.12 , regarding images see chapter 2.7.10 .

2.7 Access-i Services

2.7.1 Remote Service

2.7.1.1 Introduction

The remote service provides the RC with options to check whether and which version of the SRC is up and running on the MR host. This service requires no prior authorization of the RC.

The URL of Remote Service operation may contain the Access-i version, but it is also possible and a better approach to leave it out (as shown in the operation examples) to be able from the RC perspective to support multiple Access-i versions, i.e. finding out whether Access-i server is reachable (getIsActive) and its interface getVersion and then adapting the RC to the actual interface accordingly.

The Access-i Test Client (version 2) supports Access-i interface versions 1 and 2.

2.7.1.2 Remote Service Operations

| | | |
|-----------------------|-------|----|
| 2.7.1.2.1 getIsActive | | 27 |
| 2.7.1.2.2 getVersion | | 27 |

2.7.1.2.1 getIsActive

| | |
|--------------------|--|
| Service | remote |
| Request | getIsActive |
| Request Properties | Request Type: GET Required Privilege Level : None |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean", "isProductionSystem": "Boolean" }</pre> |
| Description | Can be used by the client to determine in general whether the server is available (independent of whether the client is registered). This command does not require a session ID as a potential client should always be able to request whether the server is available. |
| Req. Example | https://125.28.36.11:7787/SRC/product/remote/getIsActive |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608_143325.423"},"value":true,"isProductionSystem":true}</pre> |

2.7.1.2.2 getVersion

| | |
|--------------------|--|
| Service | remote |
| Request | getVersion |
| Request Properties | Request Type: GET Required Privilege Level : None |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "String" }</pre> |
| Description | Returns the version of the interface (which is currently 2.0). This command does not require a session ID. |
| Req. Example | https://125.28.36.11:7787/SRC/product/remote/getVersion |
| Resp. Exam. | <pre>{"result":{"success":true,"reason":"ok","time":"20170608_143325.423"},"value":"2.0"}</pre> |

2.7.2 Authorization Service

2.7.2.1 Introduction

The Authorization service (`authorization`) ensures that only RCs with a valid authentication key can communicate with the Access-i server. Before any of the services described in this document can be used, a RC must register providing a valid authentication key (provided by your point of contact at Siemens Healthineers), which also encodes the scope of functionality that can be used by the RC.

The Authorization service's base address format is as follows:

```
https://SERVER_IP:7787/SRC/v2/product/authorization
```

To register to the service, you have to issue the following HTTPS POST request:

```
https://SERVER_IP:7787/SRC/v2/product/authorization/register
```

The message body of this HTTPS POST request must contain the authentication key in JSON format, see highlighted in blue:

```
{ "license": { "Name": "Test Authorization
Key", "Comment": null, "WarnDate": "20180601", "StartDate": "20171109", "ExpireDate": "2018
0630", "SystemId": null, "IsReadOptionAvailable": true, "IsExecuteOptionAvailable": true,
"IsAdvancedOptionAvailable": true, "Version": "1.0", "Hash": "VX71%2BVisdUjbfz40S5f%2FRC
TKMM453ZGeQxi8jv1PdYrNWyDStu9oKVtrs%2Bv9iT9Ys4etegseQMDymkZJcaXP%2BA%3D%3D" },
"name": "Test Remote Client" }
```

The response of the register request contains the session id and privilege level that is reserved for that specific RC:

```
{ "result": { "success": true, "reason": "ok", "time": "20170608_143325.423" },
"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763",
"privilegeLevel": "advanced" }
```

For all subsequent service calls the RC must include this session id in its requests.

For HTTPS GET requests, the session id is included in the URL as in the following example:

```
https://127.0.0.1:7787/SRC/v2/product/parameter/standard/getTR?sessionId=26eb060c-
553d-4c15-bf5b-23cf29012763
```

For HTTPS POST commands, on the other hand, the session ID must be provided within the message body:

```
{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" }
```

When the RC is shut down or has no need to communicate anymore for a longer time it should deregister from the service using an HTTPS POST command:

```
https://SERVER_IP:7787/SRC/v2/product/authorization/deregister
```

Message body

```
{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" }
```

2.7.2.2 Authorization Service Operations

| | |
|---------------------------------|----|
| 2.7.2.2.1 register | 29 |
| 2.7.2.2.2 getIsRegistered | 30 |
| 2.7.2.2.3 deregister | 30 |

2.7.2.2.1 register

| | |
|--------------------|--|
| Service | authorization |
| Request | register |
| Request Properties | Request Type: POST Required Privilege Level : None |
| Parameters | <pre>{ "license": { "Name": "String", "Comment": "String", "StartDate": "String", "WarnDate": "String", "ExpireDate": "String", "SystemId": "String", "IsReadOptionAvailable": "Boolean", "IsExecuteOptionAvailable": "Boolean", "IsAdvancedOptionAvailable": "Boolean", "Version": "String", "Hash": "String" }, "name": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError otherRegisterCallInProgress noLicenseProvided licenseInvalid noClientConnectionsAllowed maintenanceActive", "time": "DateTime" }, "sessionId": "String", "privilegeLevel": "String" }</pre> |
| Description | Registering to the service using the provided license string (that represents the privilege level "Read", "Execute" or "Advanced"). Additionally, the remote client's name must be provided. |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/authorization/register</p> <p>Request body:</p> <pre>{"license":{"Name":"Access-i Authentication Key", "Comment":"Access-I Authentication Key", "StartDate":"20171212", "WarnDate":"20181201", "ExpireDate":"20181231", "SystemId":null, "IsReadOptionAvailable":true, "IsExecuteOptionAvailable":true, "IsAdvancedOptionAvailable":true, "Version":"1.0", "Hash":"sv%2BEwGRJBiWbdQ6rUbx5SIiyGyQfkEW8LDfr3leJGC5xJytXBzaf00PdMlkCxp3lVny08lAhltxdg%3D%3D"},"name":"Access-i Third Party Client"}</pre> |
| Resp. Example | <pre>{"result":{"success":true, "reason":"ok", "time":"20170608T143325.423"},"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "privilegeLevel":"advanced"}</pre> |

2.7.2.2.2 getIsRegistered

| | |
|--------------------|---|
| Service | authorization |
| Request | getIsRegistered |
| Request Properties | Request Type: GET Required Privilege Level : Standard Siemens |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError", "time": "DateTime" }, "value": "Boolean", "privilegeLevel": "String" }</pre> |
| Description | Can be used by the client to determine whether it is (still) registered. If the client is registered, the response additionally contains the client's privilege level. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/authorization/getIsRegistered?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":true,"privilegeLevel":"advanced"}</pre> |

2.7.2.2.3 deregister

| | |
|--------------------|---|
| Service | authorization |
| Request | deregister |
| Request Properties | Request Type: POST Required Privilege Level : Standard Siemens |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError", "time": "DateTime" } }</pre> |
| Description | Deregisters from the service. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/authorization/deregister Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.3 HostControl Service

2.7.3.1 Introduction

The Host Control Service allows changing the entity that is controlling the MR host which is either the MR host itself or a RC. At a given point in time only one entity can actively control the host, i.e. has the mastership with respect to controlling the workflow. As soon as a RC releases the control, the MR host is automatically in charge. The RC can only retrieve the mastership if a patient is registered and no sequence is running or open.

As soon as a RC takes over host control, a UI overlay is shown at the MR host indicating the current state (Figure 14). The overlay prevents the operating user at the MR host to execute any workflow or parameterization changes during the remote controlled phase except for the inline display viewing functionality.

However, in case of emergency or undesired RC behavior, the operating user is capable of taking over full control at the MR host, to disconnect all current RCs and prevent others to take over mastership at any time by clicking the 'Stop' button.

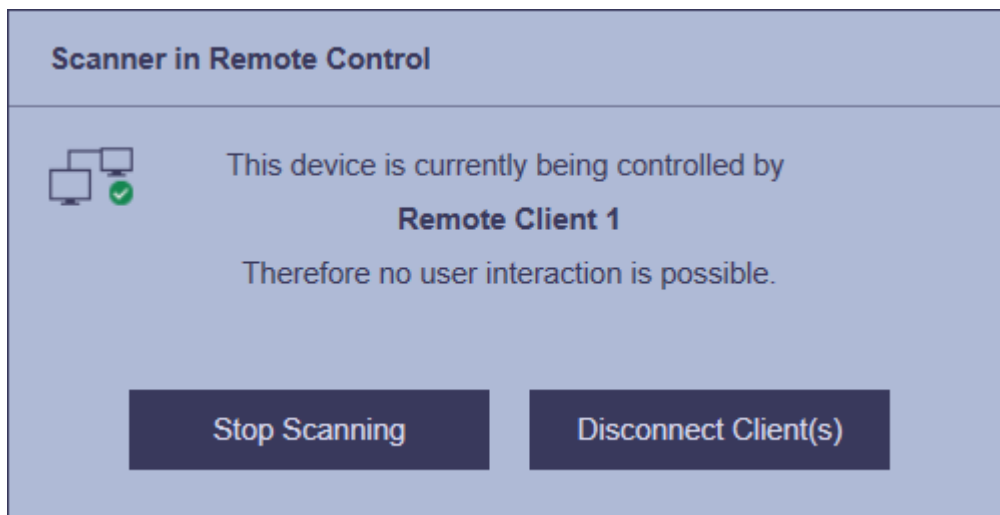


Figure 14: Example of MR host overlay when a Remote Client is in control.

2.7.3.2 HostControl Service Operations

| | |
|---|----|
| 2.7.3.2.1 getState | 33 |
| 2.7.3.2.2 requestControl | 34 |
| 2.7.3.2.3 releaseControl | 34 |
| 2.7.3.2.4 connectServiceToDefaultWebSocket..... | 35 |
| 2.7.3.2.5 connectServiceToWebSocket..... | 35 |
| 2.7.3.2.6 disconnectServiceFromDefaultWebSocket | 36 |
| 2.7.3.2.7 disconnectServiceFromWebSocket | 36 |
| 2.7.3.2.8 resendServiceStateViaWebSockets | 37 |

2.7.3.2.1 getState

| | |
|---------------------------|---|
| Service | hostControl |
| Request | getState |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "hasControl": "Boolean", "canRequestControl": "Boolean", "canReleaseControl": "Boolean", "cannotRequestControlReason": "String" } }</pre> |
| Description | <p>Returns the current state of the Host Control Service, which includes</p> <ul style="list-style-type: none"> - Has control - Can request control - Can release control - Cannot request control reason, which can be: <pre>---> clientInControl ---> noPatientRegistered ---> scanning ---> stepOpen ---> issuePending ---> anotherClientInControl</pre> |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/hostControl/getState?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"hasControl":true,"canRequestControl":false,"canReleaseControl":true,"cannotRequestControlReason":"clientInControl"}}</code> |
| WebSocket Message Example | <code>{"service":"product/hostControl","request":"getState","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"hasControl":true,"canRequestControl":false,"canReleaseControl":true,"cannotRequestControlReason":"clientInControl"}}}</code> |

2.7.3.2.2 requestControl

| | |
|--------------------|--|
| Service | hostControl |
| Request | requestControl |
| Request Properties | Request Type: POST Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError clientInControl noPatientRegistered scanning stepOpen issuePending examTaskCardNotVisible anotherClientInControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Requests control over the host |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/hostControl/requestControl Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.3.2.3 releaseControl

| | |
|--------------------|---|
| Service | hostControl |
| Request | releaseControl |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError clientNotInControl scanning stepOpen issuePending requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Releases control over the host |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/hostControl/releaseControl Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Exam. | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.3.2.4 connectServiceToDefaultWebSocket

| | |
|--------------------|--|
| Service | hostControl |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/hostControl/connectServiceToDefaultWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.3.2.5 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | hostControl |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/hostControl/connectServiceToWebSocket |

| | |
|------------------|---|
| | Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.3.2.6 disconnectServiceFromDefaultWebSocket

| | |
|-----------------------|--|
| Service | hostControl |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/hostControl/disconnectServiceFromDefaultWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.3.2.7 disconnectServiceFromWebSocket

| | |
|-----------------------|---|
| Service | hostControl |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered |

| | |
|---------------|--|
| | <pre>clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/hostControl/disconnectServiceFrom WebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.3.2.8 resendServiceStateViaWebSockets

| | |
|--------------------|---|
| Service | hostControl |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/hostControl/resendServiceStateVia WebSockets Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.3.3 Template Modification Protection

Imaging protocols which are associated with the Access-I AddIn shall enclose methods to protect against unintended modifications by the user at the MR Host. This is resolved by an overlay UI (Figure 15), which is active as soon as a RC takes over the host control. During this phase, the examination queue and thus the imaging protocols cannot be accessed and changed by the user.

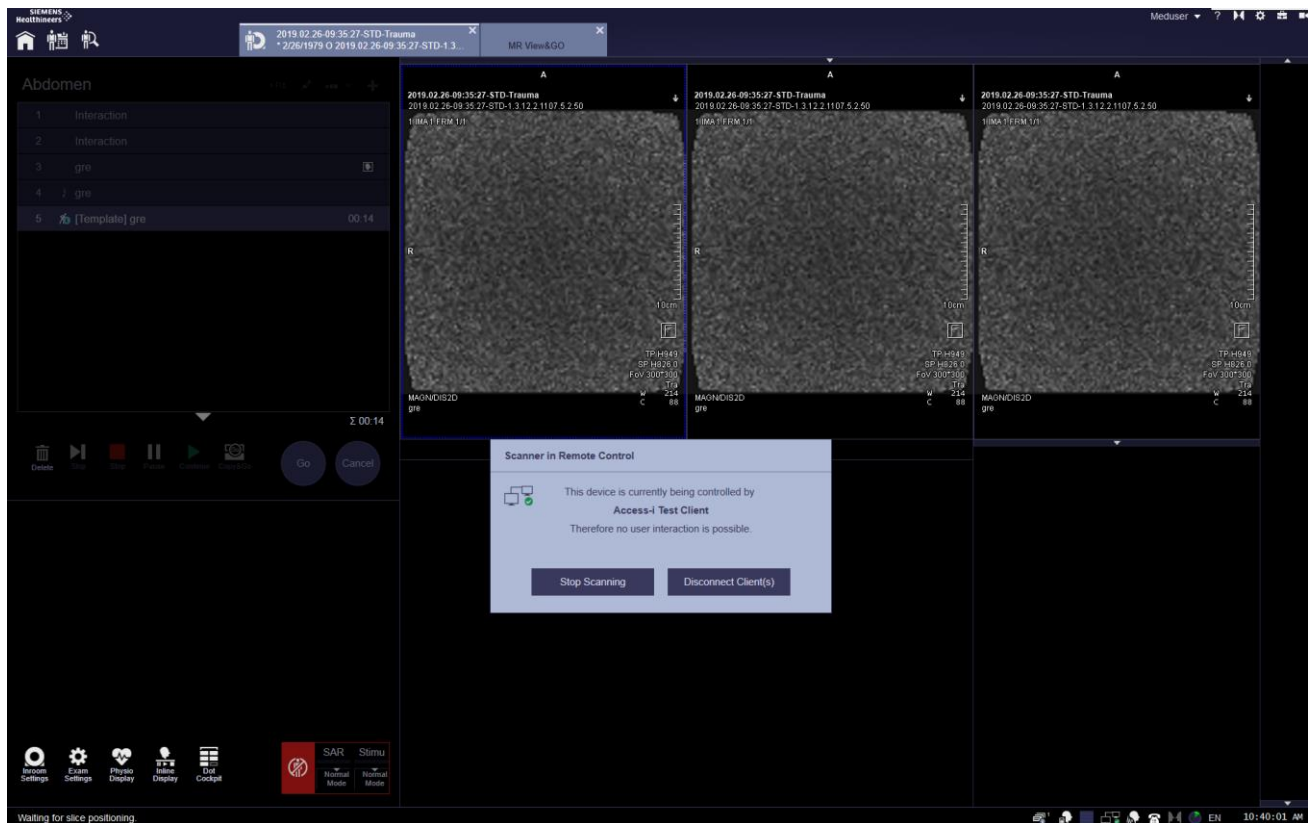


Figure 15: Overlay UI during active host control of a RC.

2.7.4 Patient Service

2.7.4.1 Introduction

The Patient Service (`patient`) allows the retrieval of the following patient-related information:

- First Name, Last Name
- PatientId
- Dicom Study Instance UID
- Dicom Study ID
- Date of birth
- Age
- Sex
- Registration direction (Head first, Feet first)
- Registration position (undefined, supine, prone, left lateral, right lateral)
- Patient language
- Height in m
- Weight in kg

2.7.4.2 Patient Service Operations

| | |
|---|----|
| 2.7.4.2.1 getPatient | 40 |
| 2.7.4.2.2 getIsRegistered | 41 |
| 2.7.4.2.3 connectServiceToDefaultWebSocket..... | 41 |
| 2.7.4.2.4 connectServiceToWebSocket..... | 42 |
| 2.7.4.2.5 disconnectServiceFromDefaultWebSocket | 42 |
| 2.7.4.2.6 disconnectServiceFromWebSocket | 43 |
| 2.7.4.2.7 resendServiceStateViaWebSockets | 44 |

2.7.4.2.1 getPatient

| | |
|---------------------------|---|
| Service | patient |
| Request | getPatient |
| Request Properties | Request Type: GET Required Privilege Level : Read Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noPatientRegistered noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "firstName": "String", "lastName": "String", "patientId": "String", "dicomStudyInstanceUid": "String", "dicomStudyId": "String", "dateOfBirth": "String", "age": "Int32", "height": "Double", "weight": "Double", "language": "String", "sex": "male female other", "direction": "headFirst feetFirst", "position": "undefined supine prone rightLateral leftLateral" } }</pre> |
| Description | Returns the patient's properties. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/patient/getPatient?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"firstName":"William","lastName":"Miller","patientId":"17.12.11-14:40:44-STD-1.3.12.2.1107.5.2.18.92097","dicomStudyInstanceUid":"1.3.12.2.1107.5.2.18.92097.3000001801081234194580000001","dicomStudyId":2,"dateOfBirth":"1999-12-11","age":18,"height":1.7,"weight":70,"language":"English","sex":"male","direction":"headFirst","position":"supine"}}</pre> |
| WebSocket Message Example | <pre>{"service":"product/patient","request":"getPatient","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"firstName":"William","lastName":"Miller","patientId":"17.12.11-14:40:44-STD-1.3.12.2.1107.5.2.18.92097","dicomStudyInstanceUid":"1.3.12.2.1107.5.2.18.92097.3000001801081234194580000001","dicomStudyId":2,"dateOfBirth":"1999-12-11","age":18,"height":1.7,"weight":70,"language":"English","sex":"male","direction":"headFirst","position":"supine"}}}</pre> |

2.7.4.2.2 getIsRegistered

| | |
|---------------------------|---|
| Service | patient |
| Request | getIsRegistered |
| Request Properties | Request Type: GET Required Privilege Level : Read Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" }</pre> |
| Description | Tells whether currently a patient is registered at the MR system |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/patient/getIsRegistered?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":true}</pre> |
| WebSocket Message Example | <pre>{"service":"product/patient","request":"getIsRegistered","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":true}}</pre> |

2.7.4.2.3 connectServiceToDefaultWebSocket

| | |
|--------------------|---|
| Service | patient |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/patient/connectServiceToDefaultWebSocket |

| | |
|---------------|--|
| | Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.4.2.4 connectServiceToWebSocket

| | |
|--------------------|--|
| Service | patient |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/patient/connectServiceToWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.4.2.5 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|--|
| Service | patient |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |

| | |
|---------------|---|
| | <pre>} }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/patient/disconnectServiceFromDefaultWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.4.2.6 disconnectServiceFromWebSocket

| | |
|--------------------|---|
| Service | patient |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/patient/disconnectServiceFromWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.4.2.7 resendServiceStateViaWebSockets

| | |
|--------------------|--|
| Service | patient |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/patient/resendServiceStateViaWebSockets Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.5 SystemInformation Service

2.7.5.1 Introduction

The System Information Service provides general information about the MR scanner. Furthermore, it provides a standardized IsoCenter position to let a Remote Client know where the system laser has been positioned.

2.7.5.2 SystemInformation Service Operations

| | |
|--|----|
| 2.7.5.2.1 getSystemInfo | 46 |
| 2.7.5.2.2 getIsoCenterPosition | 46 |
| 2.7.5.2.3 getHandballState | 47 |
| 2.7.5.2.4 getSerialNumber | 47 |
| 2.7.5.2.5 connectServiceToDefaultWebSocket..... | 48 |
| 2.7.5.2.6 connectServiceToWebSocket..... | 48 |
| 2.7.5.2.7 disconnectServiceFromDefaultWebSocket..... | 49 |
| 2.7.5.2.8 disconnectServiceFromWebSocket | 50 |
| 2.7.5.2.9 resendServiceStateViaWebSockets | 50 |

2.7.5.2.1 getSystemInfo

| | |
|--------------------|--|
| Service | systemInformation |
| Request | getSystemInfo |
| Request Properties | Request Type: GET Required Privilege Level : Basic |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "systemModel": "String", "fieldStrength": "String", "gradientType": "String", "numberOfRxChannels": "Int32", "softwareVersion": "String" } }</pre> |
| Description | Returns information about the MR system |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/systemInformation/getSystemInfo?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"systemModel":"Aera","fieldStrength":"1.5","gradientType":"XJ","numberOfRxChannels":48,"softwareVersion":"N4_VE11C_LATEST_20160120"}}</pre> |

2.7.5.2.2 getIsocenterPosition

| | |
|--------------------|---|
| Service | systemInformation |
| Request | getIsocenterPosition |
| Request Properties | Request Type: GET Required Privilege Level : Basic Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "isDefined": "Boolean", "value": "Int32" }</pre> |
| Description | Returns the Isocenter Position defined by the system laser |

| | |
|---------------------------|--|
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/systemInformation/getIsocenterPosition?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"isDefined":true,"value":132}</code> |
| WebSocket Message Example | <code>{"service":"product/systemInformation","request":"getIsocenterPosition","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"isDefined":true,"value":132}}</code> |

2.7.5.2.3 getHandballState

| | |
|---------------------------|---|
| Service | systemInformation |
| Request | getHandballState |
| Request Properties | Request Type: GET Required Privilege Level : Basic Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "idle active" }</pre> |
| Description | Returns the current state of the handball. When the handball is pressed by the patient, the state changes from 'idle' to 'active' and remains 'active' until the operator confirms this state at the communication console. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/systemInformation/getHandballState?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"idle"}</code> |
| WebSocket Message Example | <code>{"service":"product/systemInformation","request":"getHandballState","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"active"}}</code> |

2.7.5.2.4 getSerialNumber

| | |
|--------------------|---|
| Service | systemInformation |
| Request | getSerialNumber |
| Request Properties | Request Type: GET Required Privilege Level: None |
| Parameters | n/a |

| | |
|---------------|--|
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "string" }</pre> |
| Description | Returns the MR scanner's serial number. Can be called even before registering via the Authorization service to enable a third party to pick the adequate authentication key for the system. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/systemInformation/getSerialNumber</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"12345"}</code> |
| Versioning | Was newly introduced with NX-VA30A. |

2.7.5.2.5 connectServiceToDefaultWebSocket

| | |
|--------------------|---|
| Service | systemInformation |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/systemInformation/connectServiceToDefaultWebSocket</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.5.2.6 connectServiceToWebSocket

| | |
|---------|-------------------|
| Service | systemInformation |
|---------|-------------------|

| | |
|--------------------|--|
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/systemInformation/connectServiceToWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.5.2.7 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|--|
| Service | systemInformation |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/systemInformation/disconnectServiceFromDefaultWebSocket |

| | |
|---------------|---|
| | Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.5.2.8 disconnectServiceFromWebSocket

| | |
|--------------------|--|
| Service | systemInformation |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <code>{ "id": "String" }</code> |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</code> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/systemInformation/disconnectServiceFromWebSocket Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.5.2.9 resendServiceStateViaWebSockets

| | |
|--------------------|--|
| Service | systemInformation |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", } }</code> |

| | |
|---------------|---|
| | <pre>"time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/systemInformation/resendServiceStateViaWebSockets Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.5.3 Restoring Isocenter Position after it was lost due to an accidental triggering of the laser button at the scanner

2.7.5.3.1 Introduction

The Isocenter position represents the origin of the Patient Coordinate System (PCS). It is defined by turning on the system laser directly at the scanner, moving the patient table so that the laser points to the target point of interest on the patient and a longer press on the „Move to Isocenter“ button to finalize the definition process.



Image 1: The Table Control of the Siemens Skyra / Aera MR Scanners, System Laser and Isocenter Position Buttons

Alternatively, the Isocenter can also be defined by turning on the system laser, moving the table to the desired position and then turning off the system laser again.

During an exam or an intervention the Isocenter Position must not change as this leads to images that show different anatomy at the same position due to the changed origin of the PCS, i.e. images taken based on two different origins cannot be compared (different “frame of reference”).

2.7.5.3.2 Problem

In rare cases the medical staff may accidentally press the system laser button, thus invalidating and redefining a new (and arbitrary) Isocenter position. This would lead to the problem that all following scans have a different reference Isocenter position ("frame of reference"), meaning that they show a „wrong“ anatomy compared to the images taken previously.

2.7.5.3.3 Solution

To solve this problem every time the Isocenter position is defined at the scanner the absolute coordinate (relative to the "Home Position" of the table) of this newly defined position is send to the Access-i client to memorize. If the problem occurs that the Isocenter is accidentally redefined by medical staff at the scanner, this new position is also communicated to the client, so the client can detect that the Isocenter position has changed.

If this happens, the Access-i client computer should advice the user to do the following to restore the initial Isocenter position:

1. Move the table to „Home Position“



Image 2: Move to Home Position

When this button is pressed, the table completely moves out.

2. Turn the system laser on, **wait for at least 3 seconds**, and turn it off again to redefine the Isocenter at the „Home Position“

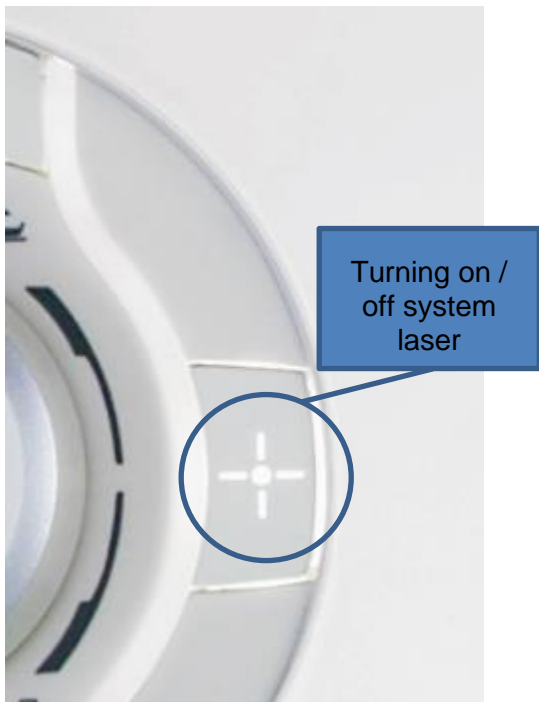


Image 3: Redefining Isocenter at a well-known position ("Home Position")

3. The user has to move the table to the position that was communicated to the Access-i client when the Isocenter was (correctly) defined at the beginning of the procedure. The user can see this value directly on the Patient Data Display at the scanner (located above the table).
4. When the table is at the correct position the user just needs to turn on the system laser again and to press the "Move to Isocenter" button to define a new Isocenter position that is identical to the original one. (Same procedure as the initial definition of the Isocenter.)

2.7.6 TemplateExecution Service

2.7.6.1 Introduction

The Template Execution Service (`templateExecution`) allows retrieving all templates that can be remotely executed, i.e. that are currently loaded in the queue. The controlling RC may start, stop, pause and continue a template and retrieve the remaining measurement of the currently running template.

NOTE: The RC is always able to stop a running sequence, also when it is not in control.

2.7.6.2 TemplateExecution Service Operations

| | |
|--|----|
| 2.7.6.2.1 getTemplates | 56 |
| 2.7.6.2.2 getState | 57 |
| 2.7.6.2.3 getRemainingMeasurementTimeInSeconds | 58 |
| 2.7.6.2.4 start | 58 |
| 2.7.6.2.5 stop | 59 |
| 2.7.6.2.6 pause | 60 |
| 2.7.6.2.7 continue | 60 |
| 2.7.6.2.8 connectServiceToDefaultWebSocket..... | 61 |
| 2.7.6.2.9 connectServiceToWebSocket..... | 61 |
| 2.7.6.2.10 disconnectServiceFromDefaultWebSocket | 62 |
| 2.7.6.2.11 disconnectServiceFromWebSocket | 62 |
| 2.7.6.2.12 resendServiceStateViaWebSockets | 63 |

2.7.6.2.1 getTemplates

| | |
|---------------------------|--|
| Service | templateExecution |
| Request | getTemplates |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": [{ "id": "Guid", "label": "String", "isInteractive": "Boolean" }] }</pre> |
| Description | Retrieves a list of available templates in the queue that can be remotely executed. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateExecution/getTemplates?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":[{"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"Interactive Template 1","isInteractive":true},{id":"abeb060c-553d-4c15-bf5b-23cf29012763","label":"T2 Template","isInteractive":false}]}</pre> |
| WebSocket Message Example | <pre>{"service":"product/templateExecution","request":"getTemplates","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":[{"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"Interactive Template 1","isInteractive":true},{id":"abeb060c-553d-4c15-bf5b-23cf29012763","label":"T2 Template","isInteractive":false}]}}</pre> |

2.7.6.2.2 getState

| | |
|---------------------------|--|
| Service | templateExecution |
| Request | getState |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "runningTemplate": { "isApplicable": "Boolean", "isTemplate": "Boolean", "isInteractive": "Boolean", "id": "Guid", "label": "String" }, "canStart": "Boolean", "canStop": "Boolean", "canPause": "Boolean", "canContinue": "Boolean", "executionState": "idle preparing issuePending scanning" } }</pre> |
| Description | <p>Returns the current state of the Template Execution Service, which includes</p> <ul style="list-style-type: none"> - Running Template - Execution State (idle, preparing, issuePending, scanning) - Can start - Can stop - Can Pause - Can Continue |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateExecution/getState?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"runningTemplate":{"isApplicable":true,"isTemplate":true,"isInteractive":true,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"Interactive Template 1"},"canStart":false,"canStop":true,"canPause":true,"canContinue":false,"executionState":"scanning"}}</pre> |
| WebSocket Message Example | <pre>{"service":"product/templateExecution","request":"getState","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"runningTemplate":{"isApplicable":true,"isTemplate":true,"isInteractive":true,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":" Interactive Template 1"},"canStart":false,"canStop":true,"canPause":true,"canContinue":false,"executionState":"scanning"}}}</pre> |

2.7.6.2.3 getRemainingMeasurementTimeInSeconds

| | |
|---------------------------|---|
| Service | templateExecution |
| Request | getRemainingMeasurementTimeInSeconds |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "isApplicable": "Boolean", "value": "Double" }</pre> |
| Description | The "isApplicable" return value tells whether a step is running and the remaining measurement time can be provided. Parameter "value" contains the remaining measurement time in seconds. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateExecution/getRemainingMeasurementTimeInSeconds?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"isApplicable":true,"value":15.5} |
| WebSocket Message Example | {"service":"product/templateExecution","request":"getRemainingMeasurementTimeInSeconds","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"isApplicable":true,"value":15.5}} |

2.7.6.2.4 start

| | |
|--------------------|---|
| Service | templateExecution |
| Request | start |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "id": "Guid" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError templateExecutionActionInProgress idInvalid interactiveTemplateNotOpen onlyOpenTemplateCanBeStarted timeout requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |

| | |
|---------------|--|
| Description | Starts template provided by the template's id. "success" is false if the template cannot be started. Prerequisites: - No other template / step is running - Interactive sequences must be open before they can be started |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/templateExecution/start</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"26eb060c-553d-4c15-bf5b-23cf29012763"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.6.2.5 stop

| | |
|--------------------|---|
| Service | templateExecution |
| Request | stop |
| Request Properties | Request Type: POST Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok unknownError templateExecutionActionInProgress noTemplateRunning timeout noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</code> |
| Description | Stops the currently running template. "success" is false if the template cannot be stopped. Prerequisites: - Template is running Stop explicitly does not require Host Control as every client that has at least "execute" privilege level should be able to stop a running sequence at any time. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/templateExecution/stop</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.6.2.6 pause

| | |
|--------------------|---|
| Service | templateExecution |
| Request | pause |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError templateExecutionActionInProgress noTemplateRunning timeout noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | <p>Pauses the currently running template. "success" is false if the template cannot be paused.</p> <p>Prerequisites:</p> <ul style="list-style-type: none"> - Template is running - Underlying sequence must be capable of being paused |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/templateExecution/pause</p> <p>Request body:</p> <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.6.2.7 continue

| | |
|--------------------|---|
| Service | templateExecution |
| Request | continue |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError templateExecutionActionInProgress noTemplateRunning timeout noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | <p>Continues the currently paused template. "success" is false if the template cannot be continued.</p> <p>Prerequisites:</p> <ul style="list-style-type: none"> - Template is running and paused |

| | |
|---------------|---|
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateExecution/continue Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.6.2.8 connectServiceToDefaultWebSocket

| | |
|--------------------|--|
| Service | templateExecution |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateExecution/connectServiceToDefaultWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.6.2.9 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | templateExecution |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered " |

| | |
|---------------|--|
| | <pre>clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/templateExecution/connectServiceToWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.6.2.10 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|---|
| Service | templateExecution |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/templateExecution/disconnectServiceFromDefaultWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.6.2.11 disconnectServiceFromWebSocket

| | |
|---------|--------------------------------|
| Service | templateExecution |
| Request | disconnectServiceFromWebSocket |
| Request | Request Type: POST |

| | |
|---------------|--|
| Properties | Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateExecution/disconnectServiceFromWebSocket Request body: <pre>{"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}}</pre> |

2.7.6.2.12 resendServiceStateViaWebSockets

| | |
|--------------------|--|
| Service | templateExecution |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateExecution/resendServiceStateViaWebSockets Request body: <pre>{"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}}</pre> |

2.7.7 TemplateSelection Service

2.7.7.1 Introduction

The Template Selection Service allows the RC to retrieve all available template programs that are remotely available to be loaded into the queue of the MR Host. The client may load these programs into the queue or just a subset of Template Steps that are part of a program. Finally, the client may clear the queue by removing the current program from the queue.

2.7.7.2 TemplateSelection Service Operations

| | |
|---|----|
| 2.7.7.2.1 connectServiceToDefaultWebSocket..... | 65 |
| 2.7.7.2.2 connectServiceToWebSocket..... | 65 |
| 2.7.7.2.3 disconnectServiceFromDefaultWebSocket | 66 |
| 2.7.7.2.4 disconnectServiceFromWebSocket | 66 |
| 2.7.7.2.5 resendServiceStateViaWebSockets | 67 |
| 2.7.7.2.6 getTemplatePrograms | 68 |
| 2.7.7.2.7 getTemplateSteps | 69 |
| 2.7.7.2.8 getLastModificationDateTime | 70 |
| 2.7.7.2.9 loadTemplateProgram | 71 |
| 2.7.7.2.10 loadTemplateSteps..... | 71 |
| 2.7.7.2.11 removeCurrentProgram | 72 |
| 2.7.7.2.12 removeTemplates | 73 |

2.7.7.2.1 connectServiceToDefaultWebSocket

| | |
|--------------------|--|
| Service | templateSelection |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/connectServiceToDefaultWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.7.2.2 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | templateSelection |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/connectServiceToWebSocket |

| | |
|---------------|---|
| | Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

2.7.7.2.3 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|--|
| Service | templateSelection |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/disconnectServiceFromDefaultWebSocket Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

2.7.7.2.4 disconnectServiceFromWebSocket

| | |
|--------------------|--|
| Service | templateSelection |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered </pre> |

| | |
|---------------|---|
| | <pre>clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/templateSelection/disconnectServiceFromWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.7.2.5 resendServiceStateViaWebSockets

| | |
|--------------------|---|
| Service | templateSelection |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/templateSelection/resendServiceStateViaWebSockets Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.7.2.6 getTemplatePrograms

| | |
|--------------------|--|
| Service | templateSelection |
| Request | getTemplatePrograms |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": [{ "id": "Guid", "path": ["String"] }] }</pre> |
| Description | Retrieve all template programs that are available on the MR Host |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/getTemplatePrograms?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":[{"id":"ae2634a3-6583-4328-9986-2f19874a0cda","path":["Root","USER","Access-i","Test","TestProgram"]}]}</pre> |

2.7.7.2.7 getTemplateSteps

| | |
|--------------------|---|
| Service | templateSelection |
| Request | getTemplateSteps |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | { "id": "Guid" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": [{ "id": "Guid", "name": "String", "programId": "Guid" }] } |
| Description | Retrieve all template steps that belong to the provided program |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/getTemplateSteps?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":[{"id":"50854324-de2c-4b4e-8d2b-62bebb3d1ded","name":"Template 1","programId":"ae2634a3-6583-4328-9986-2f19874a0cda"}]} |

2.7.7.2.8 getLastModificationDateTime

| | |
|---------------------------|---|
| Service | templateSelection |
| Request | getLastModificationDateTime |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "String" }</pre> |
| Description | Retrieve last modification datetime of the program repository |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/getLastModificationDateTime?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":"20180108T140909.326"}</pre> |
| WebSocket Message Example | <pre>{"service":"product/templateSelection","request":"getLastModificationDateTime","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"20180108T140909.326"}}</pre> |

2.7.7.2.9 loadTemplateProgram

| | |
|--------------------|--|
| Service | templateSelection |
| Request | loadTemplateProgram |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "id": "Guid" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Loads the provided template program into the queue by appending it to the existing steps that already reside in the queue. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/loadTemplateProgram Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "ae2634a3-6583-4328-9986-2f19874a0cda" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.7.2.10 loadTemplateSteps

| | |
|--------------------|--|
| Service | templateSelection |
| Request | loadTemplateSteps |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "ids": ["Guid"] } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |

| | |
|---------------|---|
| Description | Loads the provided list of steps into the queue |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/loadTemplateSteps Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "ids": ["50854324-de2c-4b4e-8d2b-62bebb3d1ded"] }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

Be aware that the static step id that is provided by `getTemplateSteps` is not the same as the id the template will have once it is loaded into the queue. Therefore it is always necessary to get an update of `templateExecution/getTemplates` after `loadTemplateSteps` in order to get the dynamic Template live-id once it is loaded into the queue. This behavior is required as the same template step can be loaded multiple times into the queue.

2.7.7.2.11 removeCurrentProgram

| | |
|--------------------|--|
| Service | templateSelection |
| Request | removeCurrentProgram |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Remove all steps that currently reside in the queue |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateSelection/removeCurrentProgram Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

2.7.7.2.12 removeTemplates

| | |
|--------------------|--|
| Service | templateSelection |
| Request | removeTemplates |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "ids": ["Guid"] }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Remove templates from the queue (identified by the provided Template live-ids). |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/templateSelection/removeTemplates</p> <p>Request body:</p> <pre>{"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "ids": ["123a60c-553d-4c15-bf5b-23cf29012763", "e28b60c-553d-4c15-bf5b-23cf29012763"]}</pre> |
| Resp. Example | <pre>{"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}}</pre> |

2.7.8 TemplateModification Service

2.7.8.1 Introduction

The Template Modification Service (`templateModification`) allows opening and closing of templates in the queue and saving changes of template parameters (close with parameter `'saveChanges' == true`). Template parameters may only be changed or queried while the template is open using the Parameter Service.

2.7.8.2 TemplateModification Service Operations

| | | |
|------------|--|----|
| 2.7.8.2.1 | <code>getState</code> | 75 |
| 2.7.8.2.2 | <code>open</code> | 76 |
| 2.7.8.2.3 | <code>close</code> | 76 |
| 2.7.8.2.4 | <code>connectServiceToDefaultWebSocket</code> | 77 |
| 2.7.8.2.5 | <code>connectServiceToWebSocket</code> | 77 |
| 2.7.8.2.6 | <code>disconnectServiceFromDefaultWebSocket</code> | 78 |
| 2.7.8.2.7 | <code>disconnectServiceFromWebSocket</code> | 79 |
| 2.7.8.2.8 | <code>resendServiceStateViaWebSockets</code> | 79 |
| 2.7.8.2.9 | <code>getLastUserEditedTemplate</code> | 80 |
| 2.7.8.2.10 | <code>setProtocolName</code> | 81 |

2.7.8.2.1 getState

| | |
|---------------------------|--|
| Service | templateModification |
| Request | getState |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "openTemplate": { "isApplicable": "Boolean", "isTemplate": "Boolean", "isInteractive": "Boolean", "id": "Guid", "label": "String" }, "canOpen": "Boolean", "canClose": "Boolean" } }</pre> |
| Description | Returns the current state of the Template Modification Service, which includes <ul style="list-style-type: none"> - Open Template - Can open - Can close |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateModification/getState?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"openTemplate":{"isApplicable":true,"isTemplate":true,"isInteractive":true,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"Interactive Template 1"},"canOpen":false,"canClose":true}}</pre> |
| WebSocket Message Example | <pre>{"service":"product/templateModification","request":"getState","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":{"openTemplate":{"isApplicable":true,"isTemplate":true,"isInteractive":true,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"Interactive Template 1"},"canOpen":false,"canClose":true}}}</pre> |

2.7.8.2.2 open

| | |
|--------------------|---|
| Service | templateModification |
| Request | open |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "id": "Guid" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError idInvalid anotherTemplateAlreadyOpen anotherTemplateModificationActionInProgress requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Opens the template with Template live-id 'id' in the queue. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateModification/open Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.8.2.3 close

| | |
|--------------------|---|
| Service | templateModification |
| Request | close |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "saveChanges": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen anotherTemplateModificationActionInProgress timeout requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Closes the currently open template and saves parameter changes if saveChanges == true or discards changes if saveChanges == false. |

| | |
|---------------|---|
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/templateModification/close</p> <p>Request body:</p> <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "saveChanges": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

2.7.8.2.4 connectServiceToDefaultWebSocket

| | |
|--------------------|---|
| Service | templateModification |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | <p>Request Type: POST</p> <p>Required Privilege Level : Read</p> |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | <p>Enables Websockets for this service</p> <p>Prerequisites:</p> <ul style="list-style-type: none"> - none |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/templateModification/connectServiceToDefaultWebSocket</p> <p>Request body:</p> <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

2.7.8.2.5 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | templateModification |
| Request | connectServiceToWebSocket |
| Request Properties | <p>Request Type: POST</p> <p>Required Privilege Level : Read</p> |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive" } }</pre> |

| | |
|---------------|---|
| | <pre>noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/templateModification/connectServiceToWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.8.2.6 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|---|
| Service | templateModification |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/templateModification/disconnectServiceFromDefaultWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.8.2.7 disconnectServiceFromWebSocket

| | |
|--------------------|--|
| Service | templateModification |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateModification/disconnectServiceFromWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.8.2.8 resendServiceStateViaWebSockets

| | |
|--------------------|--|
| Service | templateModification |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |

| | |
|---------------|--|
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateModification/resendServiceStateViaWebSockets Request body: <pre>{"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}}</pre> |

2.7.8.2.9 getLastUserEditedTemplate

| | |
|--------------------|---|
| Service | templateModification |
| Request | getLastUserEditedTemplate |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "isApplicable": "Boolean", "isTemplate": "Boolean", "isInteractive": "Boolean", "id": "Guid", "label": "String" }, "time": "String" }</pre> |
| Description | Returns the template that was last applied by the user directly at the MR Host. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateModification/getLastUserEditedTemplate?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}, "value": {"isApplicable": true, "isTemplate": true, "isInteractive": true, "id": "26eb060c-553d-4c15-bf5b-23cf29012763", "label": "Interactive Template 1"}, "time": "20170608T143001.325"}</pre> |

2.7.8.2.10 setProtocolName

| | |
|--------------------|---|
| Service | templateModification |
| Request | setProtocolName |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Sets the name of the protocol / sequence that is about to be run. Must be called while a protocol is open. If the protocol is closed and opened again before it is started, the name must be set again - otherwise the protocol template's name will be used. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/templateModification/setProtocolName Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": "Sagittal Scan" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.9 Parameter/standard Service

2.7.9.1 Introduction

The Standard Parameter Service (`parameter/standard`) allows remote access to the protocol parameters of a template that is currently open in the program queue at the scanner. This includes reading and writing several protocol parameters like acquisition order, positioning mode, slice group position and orientation, base resolution and more.

In general, in an MR protocol many parameters have dependencies among each other, meaning that changing one parameter may result in a change of one or multiple other parameters.

Therefore each protocol parameter set operation has an additional Boolean parameter `allowSideEffects`. If this is set to true, the parameter will be set to the desired value irrespective of changes of other parameters (meaning that the remote client automatically accepts all induces changes of other parameters).

If `allowSideEffects` is set to false and a change in the parameter would induce other parameter changes, the set operation fails, the protocol parameter remains unchanged and the result reason `parameterChangeWouldCauseSideEffects` will be returned to the remote client.

Furthermore - and independent of the `allowSideEffects` parameter - there are protocol parameters that only allow certain concrete values or/and have a maximum and a minimum value.

If the value to be set cannot be set exactly due to these constraints it will be set to the next possible value (meaning the next possible concrete value or - in case the value set is above the allowed maximum - to the maximum value) and the set operation succeeds.

To allow the remote client to check this incident each set operation has a `valueSet` return value that shows which value actually was set, so the remote client can see whether the supposed and actual values differ.

2.7.9.2 Parameter/standard Service Operations

| | |
|---|-----|
| 2.7.9.2.1 getTI..... | 86 |
| 2.7.9.2.2 setTI..... | 86 |
| 2.7.9.2.3 getConnectedCoils | 87 |
| 2.7.9.2.4 getIsCoilActive | 88 |
| 2.7.9.2.5 activateCoil | 88 |
| 2.7.9.2.6 deactivateCoil | 90 |
| 2.7.9.2.7 getSupportedReconstructedImageTypes | 90 |
| 2.7.9.2.8 getReconstructedImageTypes..... | 91 |
| 2.7.9.2.9 setReconstructedImageTypes..... | 91 |
| 2.7.9.2.10 getScanTimeInSeconds | 92 |
| 2.7.9.2.11 getCoilCombineMode..... | 93 |
| 2.7.9.2.12 setCoilCombineMode | 93 |
| 2.7.9.2.13 getSaveUncombined | 94 |
| 2.7.9.2.14 setSaveUncombined | 95 |
| 2.7.9.2.15 getEchoSpacing | 95 |
| 2.7.9.2.16 setEchoSpacing | 96 |
| 2.7.9.2.17 getEpiFactor..... | 96 |
| 2.7.9.2.18 setEpiFactor | 97 |
| 2.7.9.2.19 getMeasurements | 98 |
| 2.7.9.2.20 setMeasurements..... | 98 |
| 2.7.9.2.21 getDephasingGradientAngles..... | 99 |
| 2.7.9.2.22 setDephasingGradientAngles..... | 99 |
| 2.7.9.2.23 getTrackingOnly | 100 |
| 2.7.9.2.24 setTrackingOnly | 101 |
| 2.7.9.2.25 getAvailablePrimaryTrackingDevices | 101 |
| 2.7.9.2.26 getPrimaryTrackingDevice | 102 |
| 2.7.9.2.27 setPrimaryTrackingDevice..... | 102 |
| 2.7.9.2.28 getAcquisitionOrder..... | 103 |
| 2.7.9.2.29 setAcquisitionOrder | 103 |
| 2.7.9.2.30 getSkipPhysioTrigger | 104 |
| 2.7.9.2.31 setSkipPhysioTrigger | 105 |
| 2.7.9.2.32 getTracking | 105 |
| 2.7.9.2.33 setTracking | 106 |
| 2.7.9.2.34 getTrackingFlipAngle..... | 106 |

| | |
|---|-----|
| 2.7.9.2.35 setTrackingFlipAngle | 107 |
| 2.7.9.2.36 getTrackingBackgroundSuppression | 109 |
| 2.7.9.2.37 setTrackingBackgroundSuppression | 109 |
| 2.7.9.2.38 getTagSliceFollowing | 110 |
| 2.7.9.2.39 setTagSliceFollowing | 110 |
| 2.7.9.2.40 getTagSliceFollowingMode | 111 |
| 2.7.9.2.41 setTagSliceFollowingMode..... | 112 |
| 2.7.9.2.42 getTagSliceFollowingSensitivity | 112 |
| 2.7.9.2.43 setTagSliceFollowingSensitivity | 113 |
| 2.7.9.2.44 getInplaneRotationInDegrees..... | 113 |
| 2.7.9.2.45 setInplaneRotationInDegrees | 114 |
| 2.7.9.2.46 getInplaneRotationInRadians | 115 |
| 2.7.9.2.47 setInplaneRotationInRadians | 115 |
| 2.7.9.2.48 getMicroCoilDistanceToTip | 116 |
| 2.7.9.2.49 setMicroCoilDistanceToTip..... | 116 |
| 2.7.9.2.50 clearAllMicroCoilDistanceToTip..... | 117 |
| 2.7.9.2.51 getInteractiveSaturation | 118 |
| 2.7.9.2.52 setInteractiveSaturation..... | 119 |
| 2.7.9.2.53 getEcgActive | 119 |
| 2.7.9.2.54 setEcgActive | 120 |
| 2.7.9.2.55 getAvailableSequenceTypes | 120 |
| 2.7.9.2.56 getSequenceType | 121 |
| 2.7.9.2.57 setSequenceType | 122 |
| 2.7.9.2.58 getAvailableTrajectoryTypes | 123 |
| 2.7.9.2.59 getTrajectoryType | 123 |
| 2.7.9.2.60 setTrajectoryType | 124 |
| 2.7.9.2.61 getAvailablePatModes..... | 125 |
| 2.7.9.2.62 getPatMode..... | 125 |
| 2.7.9.2.63 setPatMode..... | 126 |
| 2.7.9.2.64 getT2PreparationDuration | 127 |
| 2.7.9.2.65 setT2PreparationDuration | 127 |
| 2.7.9.2.66 getSlicePositionDcs | 129 |
| 2.7.9.2.67 setSlicePositionDcs..... | 130 |
| 2.7.9.2.68 getSlicePositionPcs..... | 131 |
| 2.7.9.2.69 setSlicePositionPcs | 132 |

| | | |
|-------------|-------------------------------------|-----|
| 2.7.9.2.70 | getSliceOrientationDcs | 133 |
| 2.7.9.2.71 | setSliceOrientationDcs | 133 |
| 2.7.9.2.72 | getSliceOrientationPcs | 135 |
| 2.7.9.2.73 | setSliceOrientationPcs | 135 |
| 2.7.9.2.74 | getAdjustVolumePositionPcs..... | 137 |
| 2.7.9.2.75 | setAdjustVolumePositionPcs..... | 137 |
| 2.7.9.2.76 | getAdjustVolumeOrientationPcs | 138 |
| 2.7.9.2.77 | setAdjustVolumeOrientationPcs | 139 |
| 2.7.9.2.78 | getAdjustVolumePositionDcs | 140 |
| 2.7.9.2.79 | setAdjustVolumePositionDcs..... | 141 |
| 2.7.9.2.80 | getAdjustVolumeOrientationDcs..... | 143 |
| 2.7.9.2.81 | setAdjustVolumeOrientationDcs | 143 |
| 2.7.9.2.82 | getAdjustVolumeDimensions..... | 145 |
| 2.7.9.2.83 | setAdjustVolumeDimensions..... | 145 |
| 2.7.9.2.84 | addSliceGroup | 146 |
| 2.7.9.2.85 | deleteSliceGroup..... | 147 |
| 2.7.9.2.86 | getNumberOfSliceGroups | 147 |
| 2.7.9.2.87 | getNumberOfSlices | 148 |
| 2.7.9.2.88 | setNumberOfSlices | 148 |
| 2.7.9.2.89 | getSliceThickness | 149 |
| 2.7.9.2.90 | setSliceThickness | 149 |
| 2.7.9.2.91 | getSliceDistanceFactor | 150 |
| 2.7.9.2.92 | setSliceDistanceFactor..... | 151 |
| 2.7.9.2.93 | getFieldOfViewRead | 151 |
| 2.7.9.2.94 | setFieldOfViewRead..... | 152 |
| 2.7.9.2.95 | getFieldOfViewPhase..... | 153 |
| 2.7.9.2.96 | setFieldOfViewPhase..... | 153 |
| 2.7.9.2.97 | getFlipAngle | 154 |
| 2.7.9.2.98 | setFlipAngle | 154 |
| 2.7.9.2.99 | getBaseResolution | 155 |
| 2.7.9.2.100 | setBaseResolution | 156 |
| 2.7.9.2.101 | getPhaseResolution | 156 |
| 2.7.9.2.102 | setPhaseResolution | 157 |
| 2.7.9.2.103 | getConcatenations | 158 |
| 2.7.9.2.104 | setConcatenations | 158 |

| | |
|------------------------------------|-----|
| 2.7.9.2.105 getBandwidth | 159 |
| 2.7.9.2.106 setBandwidth..... | 159 |
| 2.7.9.2.107 getTR | 160 |
| 2.7.9.2.108 setTR | 161 |
| 2.7.9.2.109 getTE | 161 |
| 2.7.9.2.110 setTE | 162 |
| 2.7.9.2.111 getGoTolsocenter | 162 |
| 2.7.9.2.112 setGoTolsocenter..... | 163 |

2.7.9.2.1 getTI

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getTI |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" } |
| Description | Retrieves the protocol's inversion time. Unit: ms |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTI?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1.0} |

2.7.9.2.2 setTI

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setTI |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |

| | |
|---------------|---|
| Parameters | <pre>{ "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Sets new inversion time value to the protocol. Unit: ms |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTI Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 1.0, "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 1.0 }</pre> |

2.7.9.2.3 getConnectedCoils

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getConnectedCoils |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["String"] }</pre> |
| Description | Retrieves a list of receive coils that are currently connected to the scanner |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getConnectedCoils?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "value": ["Body1", "Body2"] }</pre> |

2.7.9.2.4 getIsCoilActive

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getIsCoilActive |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | { "coil": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" } |
| Description | Tells whether the provided receive coil is active. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getIsCoilActive?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&coil=Body1 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":true} |

2.7.9.2.5 activateCoil

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | activateCoil |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "String", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" } |
| Description | Activate receive coil whose name is provided as parameter. The coil to be activated must be connected. (i.e. part of list that is returned from 'getConnectedCoils') |

| | |
|---------------|---|
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/activateCoil Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":"Body1", "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":true}</pre> |

2.7.9.2.6 deactivateCoil

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | deactivateCoil |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "String", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" } |
| Description | Deactivate receive coil whose name is provided as parameter. The coil to be deactivated must be connected. (i.e. part of list that is returned from 'getConnectedCoils') |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/deactivateCoil Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": "Body1", "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423", }, "valueSet": true } |

2.7.9.2.7 getSupportedReconstructedImageTypes

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getSupportedReconstructedImageTypes |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["magnitude phase imaginary real"] } |

| | |
|--------------|--|
| | <pre>] }</pre> |
| Description | <p>Retrieves a list of basic image types that are supported by the sequence:</p> <ul style="list-style-type: none"> • magnitude • phase • imaginary • real <p>Please note that more sophisticated image types, e.g. diffusion or T1 maps are not contained within this list.</p> |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSupportedReconstructedImageTypes?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</pre> |
| Resp. Examl | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":["magnitude","phase","imaginary","real"]}</pre> |

2.7.9.2.8 getReconstructedImageTypes

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getReconstructedImageTypes |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["magnitude phase imaginary real"] }</pre> |
| Description | <p>Gets the image types that will be reconstructed when template is scanned. Please note that more sophisticated image types, e.g. diffusion or T1 maps are not contained within this list.</p> |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getReconstructedImageTypes?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</pre> |
| Resp. Exam. | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":["magnitude","phase"]}</pre> |

2.7.9.2.9 setReconstructedImageTypes

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setReconstructedImageTypes |
| Request Properties | Request Type: POST Required Privilege Level : Execute |

| | |
|---------------|---|
| | Requires Host Control |
| Parameters | <pre>{ "value": ["magnitude phase imaginary real"] }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Sets the image types to be reconstructed when this template is scanned. Please note that more sophisticated image types, e.g. diffusion or T1 maps cannot be set by this mechanism. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setReconstructedImageTypes Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b23cf29012763", "value": ["magnitude", "phase"] }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

2.7.9.2.10 getScanTimeInSeconds

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getScanTimeInSeconds |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" }</pre> |
| Description | Retrieves the time it will take to scan the template (in seconds) |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getScanTimeInSeconds?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "value": 1 }</pre> |

2.7.9.2.11 getCoilCombineMode

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getCoilCombineMode |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "sumOfSquares adaptiveCombine" }</pre> |
| Description | Gets coil combine mode, i.e. the mode how the images of the different receive coil channel elements are combined. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getCoilCombineMode?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":"sumofsquares"}</pre> |

2.7.9.2.12 setCoilCombineMode

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setCoilCombineMode |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "sumOfSquares adaptiveCombine", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "String" }</pre> |
| Description | Sets coil combine mode, i.e. the mode how the images of the different receive coil channel elements are combined. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setCoilCombine |

| | |
|---------------|--|
| | <p>Mode</p> <p>Request body:</p> <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": "sumOfSquares", "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": "sumOfSquares" }</pre> |

2.7.9.2.13 getSaveUncombined

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getSaveUncombined |
| Request Properties | <p>Request Type: GET</p> <p>Required Privilege Level : Execute</p> |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" }</pre> |
| Description | Gets the information whether created image types should be saved uncombined, i.e. an image per receive coil channel. NOTE: if this property is active, a lot of image data will be reconstructed / data load will be created. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSaveUncombined?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "value": true }</pre> |

2.7.9.2.14 setSaveUncombined

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setSaveUncombined |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Boolean", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" } |
| Description | Sets the information whether created image types should be saved uncombined, i.e. an image per receive coil channel. NOTE: if this property is active, a lot of image data will be reconstructed / data load will be created. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSaveUncombined Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": true, "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423", }, "valueSet": true } |

2.7.9.2.15 getEchoSpacing

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getEchoSpacing |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "echoSpacing": "Integer" } |

| | |
|---------------|---|
| | <pre>"value": "Double" }</pre> |
| Description | Get echo spacing |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getEchoSpacing?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1.0}</pre> |

2.7.9.2.16 setEchoSpacing

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setEchoSpacing |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Set echo spacing |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setEchoSpacing Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":1.0, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":1.0}</pre> |

2.7.9.2.17 getEpiFactor

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getEpiFactor |
| Request Properties | Request Type: GET Required Privilege Level : Execute |

| | |
|---------------|--|
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" }</pre> |
| Description | Get Epi Factor |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getEpiFactor?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1}</pre> |

2.7.9.2.18 setEpiFactor

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setEpiFactor |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Int32", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" }</pre> |
| Description | Set Epi Factor |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setEpiFactor Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":1, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":1}</pre> |

2.7.9.2.19 getMeasurements

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getMeasurements |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" }</pre> |
| Description | Get Number of Measurements |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getMeasurements?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1}</pre> |

2.7.9.2.20 setMeasurements

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setMeasurements |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Int32", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" }</pre> |
| Description | Set Number of Measurements |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setMeasurements |

| | |
|------------------|---|
| | Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 1, "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 1 }</pre> |

2.7.9.2.21 getDephasingGradientAngles

| | |
|-----------------------|--|
| Service | parameter/standard |
| Request | getDephasingGradientAngles |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["Int32"] }</pre> |
| Description | Get dephasing gradient angles of tracking acquisition spoiler gradients relative to the readout direction. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getDephasingGradientAngles?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "value": [1, 2] }</pre> |

2.7.9.2.22 setDephasingGradientAngles

| | |
|-----------------------|--|
| Service | parameter/standard |
| Request | setDephasingGradientAngles |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": ["Int32"], "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean",</pre> |

| | |
|---------------|--|
| | <pre> "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": ["Int32"] } </pre> |
| Description | Set dephasing gradient angles of tracking acquisition spoiler gradients relative to the readout direction. |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setDephasingGradientAngles</p> <p>Request body:</p> <pre> {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": [1,2], "allowSideEffects": true} </pre> |
| Resp. Example | <pre> {"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}, "valueSet": [1,2]} </pre> |

2.7.9.2.23 getTrackingOnly

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getTrackingOnly |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" } </pre> |
| Description | Get Tracking Only. If active, solely tracking projections will be acquired. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTrackingOnly?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre> {"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}, "value": true} </pre> |

2.7.9.2.24 setTrackingOnly

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setTrackingOnly |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Boolean", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" } |
| Description | Set Tracking Only. If active, solely tracking projections will be acquired. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTrackingOnly Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": true, "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423", }, "valueSet": true } |

2.7.9.2.25 getAvailablePrimaryTrackingDevices

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getAvailablePrimaryTrackingDevices |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["String"] } |

| | |
|---------------|--|
| | <pre>] }</pre> |
| Description | Retrieves the list of tracking devices of which the primary tracking device can be selected from |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAvailablePrimaryTrackingDevices?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":["ABC","ABC"]}</code> |

2.7.9.2.26 getPrimaryTrackingDevice

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getPrimaryTrackingDevice |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "String" }</pre> |
| Description | Retrieves the Primary Tracking Device configured in the protocol, i.e. device which is taken as reference for slice following mode. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getPrimaryTrackingDevice?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":"ABC"}</code> |

2.7.9.2.27 setPrimaryTrackingDevice

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setPrimaryTrackingDevice |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "String", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean",</pre> |

| | |
|---------------|---|
| | <pre> "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "String" } </pre> |
| Description | Sets the Primary Tracking Device, i.e. device which is taken as reference for slice following mode. |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setPrimaryTrackingDevice</p> <p>Request body:</p> <pre> {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":"ABC", "allowSideEffects":true} </pre> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":"ABC"} </pre> |

2.7.9.2.28 getAcquisitionOrder

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getAcquisitionOrder |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" } </pre> |
| Description | Retrieves the slice acquisition order as configured in the protocol. The number represents the indices of the slice groups, i.e. '312' means first slice group 3, and then slice groups 1 and 2 should be acquired. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAcquisitionOrder?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":312} </pre> |

2.7.9.2.29 setAcquisitionOrder

| | |
|---------|--------------------|
| Service | parameter/standard |
|---------|--------------------|

| | |
|--------------------|--|
| Request | setAcquisitionOrder |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Int32", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" } |
| Description | Sets a new acquisition order value to the protocol. The number represents the indices of the slice groups, i.e. '312' means first slice group 3, and then slice groups 1 and 2 should be acquired. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setAcquisitionOrder Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 312, "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 312 } |

2.7.9.2.30 getSkipPhysioTrigger

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getSkipPhysioTrigger |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" } |
| Description | Retrieves the "skip physio trigger" protocol parameter. If on, potential ECG or respiratory triggering is skipped despite being active. |

| | |
|--------------|--|
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSkipPhysioTrigger?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Exam. | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":true}</code> |

2.7.9.2.31 setSkipPhysioTrigger

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setSkipPhysioTrigger |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <code>{ "value": "Boolean", "allowSideEffects": "Boolean" }</code> |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" }</code> |
| Description | Turns on or turns off the "skip physio trigger" protocol parameter. If on, potential ECG or respiratory triggering is skipped despite being active. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSkipPhysioTrigger</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":true, "allowSideEffects":true}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":true}</code> |

2.7.9.2.32 getTracking

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getTracking |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <code>{ "result": {</code> |

| | |
|---------------|--|
| | <pre> "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" } </pre> |
| Description | Retrieves the information whether active tracking is active. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTracking?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":true} </pre> |

2.7.9.2.33 setTracking

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setTracking |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre> { "value": "Boolean", "allowSideEffects": "Boolean" } </pre> |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" } </pre> |
| Description | Turns active tracking on or off. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTracking Request body: <pre> {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":true, "allowSideEffects":true} </pre> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":true} </pre> |

2.7.9.2.34 getTrackingFlipAngle

| | |
|---------|--------------------|
| Service | parameter/standard |
|---------|--------------------|

| | |
|--------------------|--|
| Request | getTrackingFlipAngle |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" } |
| Description | Retrieves the flip angle used for active tracking. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTrackingFlipAngle?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1} |

2.7.9.2.35 setTrackingFlipAngle

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setTrackingFlipAngle |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Int32", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" } |
| Description | Sets the flip angle for active tracking. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTrackingFlipAngle Request body: { "sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":1, "allowSideEffects":true } |

| | |
|------------------|---|
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":1}</pre> |
|------------------|---|

2.7.9.2.36 getTrackingBackgroundSuppression

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getTrackingBackgroundSuppression |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" }</pre> |
| Description | Retrieves background suppression strength (dephaser strength) for tracking acquisition in mT/m. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTrackingBackgroundSuppression?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1}</pre> |

2.7.9.2.37 setTrackingBackgroundSuppression

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setTrackingBackgroundSuppression |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Int32", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" }</pre> |
| Description | Sets background suppression strength (dephaser strength) for tracking acquisition in mT/m. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTrackingBackgroundSuppression |

| | |
|---------------|--|
| | kgroundSuppression Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 1, "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423"}, "valueSet": 1 } |

2.7.9.2.38 getTagSliceFollowing

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getTagSliceFollowing |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "off onSliceGroup1 onAllSliceGroups" } |
| Description | Retrieves the type set for slice following. <ul style="list-style-type: none">- Off: Slice following is deactivated- onSliceGroup1: Only the position of slice 1 follows the primary tracking device- onAllSliceGroups: The position of all slices follow the primary tracking device |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTagSliceFollowing?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423"}, "value": "off" } |

2.7.9.2.39 setTagSliceFollowing

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setTagSliceFollowing |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "off onSliceGroup1 onAllSliceGroups", "allowSideEffects": "Boolean" } |

| | |
|---------------|---|
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "String" }</pre> |
| Description | <p>Sets the type for slice following.</p> <ul style="list-style-type: none"> - Off: Slice following is deactivated - onSliceGroup1: Only the position of slice 1 follows the the primary tracking device position - onAllSliceGroups: The position of all slices follow the primary tracking device position |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTagSliceFollowing</p> <p>Request body:</p> <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":"off", "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":"ABC"}</pre> |

2.7.9.2.40 getTagSliceFollowingMode

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getTagSliceFollowingMode |
| Request Properties | <p>Request Type: GET</p> <p>Required Privilege Level : Execute</p> |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "tipTracking perpendicular parallel threePointPlane" }</pre> |
| Description | Retrieves the mode used for slice following. |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTagSliceFollowingMode?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</p> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":"tiptracking"}</pre> |

2.7.9.2.41 setTagSliceFollowingMode

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setTagSliceFollowingMode |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "tipTracking perpendicular parallel threePointPlane", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "String" } |
| Description | Sets the mode to be used for slice following. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTagSliceFollowingMode Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": "tiptracking", "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": "ABC" } |

2.7.9.2.42 getTagSliceFollowingSensitivity

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getTagSliceFollowingSensitivity |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" } |

| | |
|---------------|---|
| Description | Retrieves the slice following sensitivity in mm, i.e. the threshold when image position shall be adjusted to the primary device position |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTagSliceFollowingSensitivity?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":1}</code> |

2.7.9.2.43 setTagSliceFollowingSensitivity

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setTagSliceFollowingSensitivity |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Int32", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" }</pre> |
| Description | Sets the slice following sensitivity in mm, i.e. the threshold when image position shall be adjusted to the primary device position. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTagSliceFollowingSensitivity</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":1, "allowSideEffects":true}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"valueSet":1}</code> |

2.7.9.2.44 getInplaneRotationInDegrees

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getInplaneRotationInDegrees |
| Request Properties | Request Type: GET Required Privilege Level : Execute |

| | |
|---------------|---|
| Parameters | <pre>{ "index": "Int32" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" }</pre> |
| Description | Retrieves the inplane rotation in degrees of slice group at index "index". Unit: degrees |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getInplaneRotationInDegrees?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&index=1 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1.0}</pre> |

2.7.9.2.45 setInplaneRotationInDegrees

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setInplaneRotationInDegrees |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "index": "Int32", "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Sets the inplane rotation in degrees of slice group at index "index". Unit: degrees |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setInplaneRotationInDegrees Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "index":1, "value":1.0,</pre> |

| | |
|------------------|---|
| | "allowSideEffects":true} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":1.0} |

2.7.9.2.46 getInplaneRotationInRadians

| | |
|-----------------------|---|
| Service | parameter/standard |
| Request | getInplaneRotationInRadians |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | { "index": "Int32" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" } |
| Description | Retrieves the inplane rotation in radians of slice group at index "index" Unit: Radians |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getInplaneRotationInRadians?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&index=1 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1.0} |

2.7.9.2.47 setInplaneRotationInRadians

| | |
|-----------------------|---|
| Service | parameter/standard |
| Request | setInplaneRotationInRadians |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "index": "Int32", "value": "Double", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered |

| | |
|---------------|---|
| | <pre> clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" } </pre> |
| Description | Sets the inplane rotation in degrees of slice group at index "index". Unit: radians |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setInplaneRotationInRadians</p> <p>Request body:</p> <pre> {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "index":1, "value":1.0, "allowSideEffects":true} </pre> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":1.0} </pre> |

2.7.9.2.48 getMicroCoilDistanceToTip

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getMicroCoilDistanceToTip |
| Request Properties | <p>Request Type: GET</p> <p>Required Privilege Level : Execute</p> |
| Parameters | <pre> { "id": "String" } </pre> |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" } </pre> |
| Description | Retrieves the distance between micro coil named "id" of the tracking device and the device's tip. |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getMicroCoilDistanceToTip?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=d65f21db-e69b-49f4-830b-117cc02996ff</p> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1.0} </pre> |

2.7.9.2.49 setMicroCoilDistanceToTip

| | |
|---------|---------------------------|
| Service | parameter/standard |
| Request | setMicroCoilDistanceToTip |

| | |
|--------------------|---|
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "id": "String", "value": "Double", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" } |
| Description | Sets the distance between micro coil named "id" of the primary tracking device and the device's tip. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setMicroCoilDistanceToTip Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff", "value": 1.0, "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 1.0 } |

2.7.9.2.50 clearAllMicroCoilDistanceToTip

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | clearAllMicroCoilDistanceToTip |
| Request Properties | Request Type: POST Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Clears previously defined distances to tip for all micro coils of the primary tracking device. |

| | |
|---------------|--|
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/clearAllMicroCoilDistanceToTip Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.9.2.51 getInteractiveSaturation

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getInteractiveSaturation |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" }</pre> |
| Description | Retrieves the information from protocol whether interactive saturation is active, i.e. whether the selected magnetization preparation is switched on/off. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getInteractiveSaturation?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":true} |

2.7.9.2.52 setInteractiveSaturation

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setInteractiveSaturation |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Boolean", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" } |
| Description | Turns interactive saturation on or off, i.e. the selected magnetization preparation. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setInteractiveSaturation Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": true, "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423", }, "valueSet": true } |

2.7.9.2.53 getEcgActive

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getEcgActive |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" } |

| | |
|---------------|--|
| Description | Retrieves the information whether ECG triggering is active. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getEcgActive?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":true}</code> |

2.7.9.2.54 setEcgActive

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setEcgActive |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <code>{ "value": "Boolean", "allowSideEffects": "Boolean" }</code> |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" }</code> |
| Description | Turns ECG triggering at scanner on or off. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setEcgActive</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":true, "allowSideEffects":true}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"valueSet":true}</code> |

2.7.9.2.55 getAvailableSequenceTypes

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getAvailableSequenceTypes |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { |

| | |
|---------------|---|
| | <pre> "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["String"] } </pre> |
| Description | <p>Retrieves the list of available sequence types for the sequence of the currently open template which is a subset of</p> <ul style="list-style-type: none"> - gre - trufi - epi - tse - csi |
| Req. Example | <pre> https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAvailableSe quenceTypes?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 </pre> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":["gre","trufi"]} </pre> |

2.7.9.2.56 getSequenceType

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getSequenceType |
| Request Properties | <p>Request Type: GET</p> <p>Required Privilege Level : Execute</p> |
| Parameters | n/a |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "gre trufi epi tse csi" } </pre> |
| Description | Retrieves the sequence type configured in the protocol. |
| Req. Example | <pre> https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSequenceTyp e?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 </pre> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":"gre"} </pre> |

2.7.9.2.57 setSequenceType

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setSequenceType |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "gre trufi epi tse csi", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "String" } |
| Description | Sets the sequence type to be used. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSequenceType Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": "gre", "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": "gre" } |

2.7.9.2.58 getAvailableTrajectoryTypes

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getAvailableTrajectoryTypes |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["String"] }</pre> |
| Description | <p>Retrieves the list of available k-space trajectory types, which can be used by the protocol for image acquisition</p> <p>Possible values:</p> <ul style="list-style-type: none"> - cartesian - blade - radial - spiral |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAvailableTrajectoryTypes?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":["cartesian","blade"]}</pre> |

2.7.9.2.59 getTrajectoryType

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getTrajectoryType |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "cartesian blade radial spiral" }</pre> |
| Description | Retrieves the k-space trajectory type configured in the protocol for image |

| | |
|---------------|---|
| | acquisition. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTrajectoryType?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"cartesian"}</code> |

2.7.9.2.60 setTrajectoryType

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setTrajectoryType |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "cartesian blade radial spiral", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "String" }</pre> |
| Description | Sets the k-space trajectory type to be used for image acquisition. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTrajectoryType</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":"cartesian", "allowSideEffects":true}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"valueSet":"cartesian"}</code> |

2.7.9.2.61 getAvailablePatModes

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getAvailablePatModes |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["String"] }</pre> |
| Description | <p>Retrieves the list of available PAT Modes, i.e. techniques for parallel imaging.</p> <p>Possible values:</p> <ul style="list-style-type: none"> - none - grappa - sense - mode2D |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAvailablePatModes?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":["none","grappa"]}</pre> |

2.7.9.2.62 getPatMode

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getPatMode |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "none grappa sense mode2D" }</pre> |
| Description | Retrieves the PAT mode configured in the protocol for parallel imaging. |

| | |
|---------------|--|
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getPatMode?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"sense"}</code> |

2.7.9.2.63 setPatMode

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setPatMode |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <code>{ "value": "none grappa sense mode2D", "allowSideEffects": "Boolean" }</code> |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "String" }</code> |
| Description | Sets the PAT mode to be used for parallel imaging. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setPatMode</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":"grappa", "allowSideEffects":true}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"valueSet":"grappa"}</code> |

2.7.9.2.64 getT2PreparationDuration

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getT2PreparationDuration |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" }</pre> |
| Description | Retrieves T2 preparation duration in ms. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getT2PreparationDuration?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1.0}</pre> |

2.7.9.2.65 setT2PreparationDuration

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setT2PreparationDuration |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Sets new T2 preparation duration in ms. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setT2PreparationDuration |

| | |
|------------------|---|
| | Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 1.0, "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 1.0 }</pre> |

2.7.9.2.66 getSlicePositionDcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getSlicePositionDcs |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | { "index": "Int32" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "x": "Double", "y": "Double", "z": "Double" } } |
| Description | Retrieves the position of slice group at index "index" in device coordinates (DCS). Please note: The position information returned by getSlicePositionDcs denotes the slice group center. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSlicePositionDcs?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&index=0 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":{"x":10.0,"y":10.0,"z":20.0}} |

2.7.9.2.67 setSlicePositionDcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setSlicePositionDcs |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "index": "Int32", "value": { "x": "Double", "y": "Double", "z": "Double" }, "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": { "x": "Double", "y": "Double", "z": "Double" } }</pre> |
| Description | <p>Sets the position of slice group at index "index" in device coordinates (DCS).</p> <p>Please note: The position information provided to the setSlicePositionDcs denotes the slice group center.</p> |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSlicePositionDcs</p> <p>Request body:</p> <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "index":0, "value":{"x":10.0,"y":10.0,"z":20.0}, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":{"x":10.0,"y":10.0,"z":20.0}}</pre> |

2.7.9.2.68 getSlicePositionPcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getSlicePositionPcs |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | { "index": "Int32" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "sag": "Double", "cor": "Double", "tra": "Double" } } |
| Description | Retrieves the position of slice group at index "index" in patient coordinates (PCS). Please note: The position information returned by getSlicePositionPcs denotes the slice group center. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSlicePositionPcs?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&index=0 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":{"sag":10.0,"cor":-10.0,"tra":-20.0}} |

2.7.9.2.69 setSlicePositionPcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setSlicePositionPcs |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "index": "Int32", "value": { "sag": "Double", "cor": "Double", "tra": "Double" }, "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": { "sag": "Double", "cor": "Double", "tra": "Double" } }</pre> |
| Description | <p>Sets the position of slice group at index "index" in patient coordinates (PCS).</p> <p>Please note: The position information provided to the setSlicePositionPcs denotes the slice group center.</p> |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSlicePositionPcs</p> <p>Request body:</p> <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "index":0, "value":{"sag":10.0,"cor":-10.0,"tra":-20.0}, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":{"sag":10.0,"cor":-10.0,"tra":-20.0}}</pre> |

2.7.9.2.70 getSliceOrientationDcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getSliceOrientationDcs |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | { "index": "Int32" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normal": { "x": "Double", "y": "Double", "z": "Double" }, "phase": { "x": "Double", "y": "Double", "z": "Double" }, "read": { "x": "Double", "y": "Double", "z": "Double" } } |
| Description | Retrieves orientation (i.e. normal), phase and readout directions of slice group at index "index" in device coordinates (DCS). |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSliceOrientationDcs?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&index=0 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normal":{"x":0,"y":0,"z":-1.0},"phase":{"x":0,"y":-1.0,"z":0}, "read":{"x":-1.0,"y":0,"z":0}} |

2.7.9.2.71 setSliceOrientationDcs

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setSliceOrientationDcs |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "index": "Int32", "normal": { "x": "Double", "y": "Double", } |

| | |
|--------------|---|
| | <pre> "z": "Double" }, "phase": { "x": "Double", "y": "Double", "z": "Double" }, "read": { "x": "Double", "y": "Double", "z": "Double" }, "allowSideEffects": "Boolean" } </pre> |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normalSet": { "x": "Double", "y": "Double", "z": "Double" }, "phaseSet": { "x": "Double", "y": "Double", "z": "Double" }, "readSet": { "x": "Double", "y": "Double", "z": "Double" } } </pre> |
| Description | <p>Sets the orientation (i.e. normal), phase and readout directions of the slice group at index "index" in device coordinates (DCS).</p> <p>Please note: The vectors passed as arguments to setSliceOrientationDcs must be</p> <ul style="list-style-type: none"> - normalized (vector length == 1) - conventionalized (the largest component of a vector must be positive) - the normal, phase and read vectors must all be orthogonal to each other and normal = phase x read <p>Note: There is an in-plane half pixel deviation between the planned and resulting image center. For further details please refer to Ref. [2].</p> |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSliceOrientationDcs</p> <p>Request body:</p> <pre> {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "index":0, "normal":{"x":0,"y":0,"z":-1.0}, "phase":{"x":0,"y":-1.0,"z":0}, "read":{"x":-1.0,"y":0,"z":0}, "allowSideEffects":true} </pre> |

| | |
|---------------|---|
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normalSet":{"x":0,"y":0,"z":-1.0},"phaseSet":{"x":0,"y":-1.0,"z":0}, "readSet":{"x":-1.0,"y":0,"z":0}}</pre> |
|---------------|---|

2.7.9.2.72 getSliceOrientationPcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getSliceOrientationPcs |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | <pre>{ "index": "Int32" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normal": { "sag": "Double", "cor": "Double", "tra": "Double" }, "phase": { "sag": "Double", "cor": "Double", "tra": "Double" }, "read": { "sag": "Double", "cor": "Double", "tra": "Double" } }</pre> |
| Description | Retrieves orientation (i.e. normal), phase and readout directions of slice group at index "index" in patient coordinates (PCS). |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSliceOrientationPcs?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&index=0 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normal":{"sag":0,"cor":0,"tra":1.0},"phase":{"sag":0,"cor":1.0,"tra":0}, "read":{"sag":-1.0,"cor":0,"tra":0}}</pre> |

2.7.9.2.73 setSliceOrientationPcs

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setSliceOrientationPcs |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |

| | |
|--------------|--|
| Parameters | <pre>{ "index": "Int32", "normal": { "sag": "Double", "cor": "Double", "tra": "Double" }, "phase": { "sag": "Double", "cor": "Double", "tra": "Double" }, "read": { "sag": "Double", "cor": "Double", "tra": "Double" }, "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normalSet": { "sag": "Double", "cor": "Double", "tra": "Double" }, "phaseSet": { "sag": "Double", "cor": "Double", "tra": "Double" }, "readSet": { "sag": "Double", "cor": "Double", "tra": "Double" } }</pre> |
| Description | <p>Sets the orientation (i.e. normal), phase and readout directions of the slice group at index "index" in patient coordinates (PCS).</p> <p>Please note: The vectors passed as arguments to setSliceOrientationPcs must be - normalized (vector length == 1) - conventionalized (the largest component of a vector must be positive) - the normal, phase and read vectors must all be orthogonal to each other and normal = phase x read</p> <p>Note: There is an in-plane half pixel deviation between the planned and resulting image center. For further details please refer to Ref. [2].</p> |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSliceOrientationPcs</p> <p>Request body: {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "index": 0,</p> |

| | |
|---------------|--|
| | <pre>"normal":{"sag":0,"cor":0,"tra":1.0}, "phase":{"sag":0,"cor":1.0,"tra":0}, "read":{"sag":-1.0,"cor":0,"tra":0}, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normalSet":{"sag":0,"cor":0,"tra":1.0}"phaseSet":{"sag":0,"cor":1.0,"tra":0}, "readSet":{"sag":-1.0,"cor":0,"tra":0}}</pre> |

2.7.9.2.74 getAdjustVolumePositionPcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getAdjustVolumePositionPcs |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "sag": "Double", "cor": "Double", "tra": "Double" } }</pre> |
| Description | Retrieves the position of the Adjust Volume in patient coordinates (PCS). Please note: The position information returned by getAdjustVolumePositionPcs denotes the adjust volume center. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAdjustVolumePositionPcs?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":{"sag":30.0,"cor":40.0,"tra":50.0}}</pre> |

2.7.9.2.75 setAdjustVolumePositionPcs

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setAdjustVolumePositionPcs |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": { "sag": "Double", "cor": "Double",</pre> |

| | |
|---------------|--|
| | <pre> "tra": "Double" }, "allowSideEffects": "Boolean" } </pre> |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": { "sag": "Double", "cor": "Double", "tra": "Double" } } </pre> |
| Description | <p>Sets the position of the adjustment volume in patient coordinates (PCS).</p> <p>Please note: The position information provided to the <code>setAdjustVolumePositionPcs</code> denotes the adjust volume center.</p> |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setAdjustVolumePositionPcs</p> <p>Request body:</p> <pre> {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": {"sag": 30.0, "cor": 40.0, "tra": 50.0}, "allowSideEffects": true} </pre> |
| Resp. Example | <pre> {"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}, "valueSet": {"sag": 30.0, "cor": 40.0, "tra": 50.0}} </pre> |

2.7.9.2.76 getAdjustVolumeOrientationPcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getAdjustVolumeOrientationPcs |
| Request Properties | <p>Request Type: GET</p> <p>Required Privilege Level : Execute</p> |
| Parameters | n/a |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normal": { "sag": "Double", "cor": "Double", "tra": "Double" } }, </pre> |

| | |
|---------------|---|
| | <pre> "phase": { "sag": "Double", "cor": "Double", "tra": "Double" }, "read": { "sag": "Double", "cor": "Double", "tra": "Double" } } </pre> |
| Description | Retrieves orientation (i.e. normal), phase and readout directions of the Adjust Volume in patient coordinates (PCS). |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAdjustVolumeOrientationPcs?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normal":{"sag":0,"cor":1.0,"tra":0},"phase":{"sag":1.0,"cor":0,"tra":0}, "read":{"sag":0,"cor":0,"tra":-1.0}} </pre> |

2.7.9.2.77 setAdjustVolumeOrientationPcs

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setAdjustVolumeOrientationPcs |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre> { "normal": { "sag": "Double", "cor": "Double", "tra": "Double" }, "phase": { "sag": "Double", "cor": "Double", "tra": "Double" }, "read": { "sag": "Double", "cor": "Double", "tra": "Double" }, "allowSideEffects": "Boolean" } </pre> |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normalSet": { "sag": "Double", "cor": "Double", "tra": "Double" } } </pre> |

| | |
|---------------|--|
| | <pre> "tra": "Double" }, "phaseSet": { "sag": "Double", "cor": "Double", "tra": "Double" }, "readSet": { "sag": "Double", "cor": "Double", "tra": "Double" } } } </pre> |
| Description | <p>Sets the orientation (i.e. normal), phase and readout directions of the Adjust Volume in patient coordinates (PCS).</p> <p>Please note:</p> <ul style="list-style-type: none"> The vectors passed as arguments to setAdjustVolumeOrientationPcs must be <ul style="list-style-type: none"> normalized (vector length == 1) conventionalized (the largest component of a vector must be positive) the normal, phase and read vectors must all be orthogonal to each other and normal = phase x read |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setAdjustVolumeOrientationPcs</p> <p>Request body:</p> <pre> {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "normal":{"sag":0,"cor":1.0,"tra":0}, "phase":{"sag":1.0,"cor":0,"tra":0}, "read":{"sag":0,"cor":0,"tra":-1.0}, "allowSideEffects":true} </pre> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normalSet":{"sag":0,"cor":1.0,"tra":0}, "phaseSet":{"sag":1.0,"cor":0,"tra":0}, "readSet":{"sag":0,"cor":0,"tra":-1.0}} </pre> |

2.7.9.2.78 getAdjustVolumePositionDcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getAdjustVolumePositionDcs |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": { "sag": "Double", "cor": "Double", "tra": "Double" } } </pre> |

| | |
|--------------|---|
| | } |
| Description | Retrieves the position of the Adjust Volume in device coordinates (DCS). Please note: The position information returned by getAdjustVolumePositionDcs denotes the adjust volume center. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAdjustVolumePositionDcs?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Exam. | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":{"sag":30.0,"cor":-40.0,"tra":-50.0}}</code> |

2.7.9.2.79 setAdjustVolumePositionDcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setAdjustVolumePositionDcs |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": { "sag": "Double", "cor": "Double", "tra": "Double" }, "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": { "sag": "Double", "cor": "Double", "tra": "Double" } }</pre> |
| Description | Sets the position of the adjustment volume in device coordinates (DCS). Please note: The position information provided to the setAdjustVolumePositionDcs denotes the adjust volume center. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setAdjustVolumePositionDcs</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":{"sag":30.0,"cor":-40.0,"tra":-50.0}, "allowSideEffects":true}</code> |

| | |
|------------------|--|
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":{"sag":30.0,"cor":-40.0,"tra":-50.0}}</pre> |
|------------------|--|

2.7.9.2.80 getAdjustVolumeOrientationDcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getAdjustVolumeOrientationDcs |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normal": { "sag": "Double", "cor": "Double", "tra": "Double" }, "phase": { "sag": "Double", "cor": "Double", "tra": "Double" }, "read": { "sag": "Double", "cor": "Double", "tra": "Double" } }</pre> |
| Description | Retrieves orientation (i.e. normal), phase and readout directions of the Adjust Volume in device coordinates (DCS). |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAdjustVolumeOrientationDcs?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normal":{"sag":0,"cor":-1.0,"tra":0},"phase":{"sag":1.0,"cor":0,"tra":0}, "read":{"sag":0,"cor":0,"tra":1.0}}</pre> |

2.7.9.2.81 setAdjustVolumeOrientationDcs

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setAdjustVolumeOrientationDcs |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "normal": { "sag": "Double", "cor": "Double", "tra": "Double" }, "phase": {</pre> |

| | |
|---------------|--|
| | <pre> "sag": "Double", "cor": "Double", "tra": "Double" }, "read": { "sag": "Double", "cor": "Double", "tra": "Double" }, "allowSideEffects": "Boolean" } </pre> |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normalSet": { "sag": "Double", "cor": "Double", "tra": "Double" }, "phaseSet": { "sag": "Double", "cor": "Double", "tra": "Double" }, "readSet": { "sag": "Double", "cor": "Double", "tra": "Double" } } </pre> |
| Description | <p>Sets the orientation (i.e. normal), phase and readout directions of the Adjust Volume in patient coordinates (PCS).</p> <p>Please note:</p> <ul style="list-style-type: none"> The vectors passed as arguments to setAdjustVolumeOrientationPcs must be <ul style="list-style-type: none"> normalized (vector length == 1) conventionalized (the largest component of a vector must be positive) the normal, phase and read vectors must all be orthogonal to each other and normal = phase x read |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setAdjustVolumeOrientationDcs</p> <p>Request body:</p> <pre> {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "normal":{"sag":0,"cor":-1.0,"tra":0}, "phase":{"sag":1.0,"cor":0,"tra":0}, "read":{"sag":0,"cor":0,"tra":1.0}, "allowSideEffects":true} </pre> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normalSet":{"sag":0,"cor":-1.0,"tra":0}, "phaseSet":{"sag":1.0,"cor":0,"tra":0}, "readSet":{"sag":0,"cor":0,"tra":1.0}} </pre> |

2.7.9.2.82 getAdjustVolumeDimensions

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getAdjustVolumeDimensions |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normal": "Double", "phase": "Double", "read": "Double" }</pre> |
| Description | Retrieves the dimensions of the adjust volume in phase, read and normal direction. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getAdjustVolumeDimensions?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "normal":35.0,"phase":45.0,"read":55.0}</pre> |

2.7.9.2.83 setAdjustVolumeDimensions

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setAdjustVolumeDimensions |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "normal": "Double", "phase": "Double", "read": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "normalSet": "Double", "phaseSet": "Double", "readSet": "Double" }</pre> |

| | |
|---------------|---|
| | } |
| Description | Sets the dimensions of the adjust volume in phase, read and normal direction |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setAdjustVolumeDimensions Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "normal": 35.0, "phase": 45.0, "read": 55.0, "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "normalSet": 35.0, "phaseSet": 45.0, "readSet": 55.0 }</pre> |

2.7.9.2.84 addSliceGroup

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | addSliceGroup |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" }</pre> |
| Description | Adds one slice group to the protocol and returns the new number of slice groups available in the protocol. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/addSliceGroup Request body: <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 2 }</pre> |

2.7.9.2.85 deleteSliceGroup

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | deleteSliceGroup |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" } |
| Description | Deletes one slice group from the protocol and returns the new number of slice groups available in the protocol. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/deleteSliceGroup Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423", }, "valueSet": 1 } |

2.7.9.2.86 getNumberOfSliceGroups

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getNumberOfSliceGroups |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" } |
| Description | Retrieves the number of slice groups in the protocol. |

| | |
|---------------|---|
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getNumberOfSlices?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":2}</code> |

2.7.9.2.87 getNumberOfSlices

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getNumberOfSlices |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | <code>{ "index": "Int32" }</code> |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" }</code> |
| Description | Retrieves the number of slices of slice group at index "index". |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getNumberOfSlices?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&index=0</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":3}</code> |

2.7.9.2.88 setNumberOfSlices

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setNumberOfSlices |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <code>{ "index": "Int32", "value": "Int32", "allowSideEffects": "Boolean" }</code> |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl" } }</code> |

| | |
|---------------|--|
| | <pre> noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" }</pre> |
| Description | Sets the number of slices of slice group at index "index". |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setNumberOfSlices</p> <p>Request body:</p> <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "index": 0, "value": 4, "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 4 }</pre> |

2.7.9.2.89 getSliceThickness

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getSliceThickness |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" }</pre> |
| Description | Retrieves the slice thickness configured in the protocol for all slice groups. Unit: mm |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSliceThickness?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "value": 2.5 }</pre> |

2.7.9.2.90 setSliceThickness

| | |
|---------|--|
| Service | parameter/standard |
| Request | setSliceThickness |
| Request | Request Type: POST Required Privilege Level : Execute |

| | |
|---------------|---|
| Properties | Requires Host Control |
| Parameters | <pre>{ "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Sets the slice thickness for all slice groups in the protocol. Unit: mm |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSliceThickness Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":3.5, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":3.5}</pre> |

2.7.9.2.91 getSliceDistanceFactor

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getSliceDistanceFactor |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | <pre>{ "index": "Int32" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" }</pre> |
| Description | Retrieves the slice distance factor of the slice group at index 'index'. Unit: 0..n where 1 represents 100% in the protocol |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getSliceDistanceFactor?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&index=0 |

| | |
|---------------|--|
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":0.4}</pre> |
|---------------|--|

2.7.9.2.92 setSliceDistanceFactor

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setSliceDistanceFactor |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "index": "Int32", "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Sets the slice distance factor of the slice group at index 'index'. Unit: 0..n where 1 represents 100% in the protocol |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setSliceDistan ceFactor Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "index":0, "value":0.6, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":0.6}</pre> |

2.7.9.2.93 getFieldOfViewRead

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getFieldOfViewRead |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": {</pre> |

| | |
|---------------|---|
| | <pre> "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" } </pre> |
| Description | Retrieves the field of view dimension in readout direction configured in the protocol for all slice groups (there is only one field of view that applies to all slice groups in the protocol). Unit: mm |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getFieldOfViewRead?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":400.0} </pre> |

2.7.9.2.94 setFieldOfViewRead

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setFieldOfViewRead |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre> { "value": "Double", "allowSideEffects": "Boolean" } </pre> |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" } </pre> |
| Description | Sets the field of view dimension in readout direction for to all slice groups in the protocol. Unit: mm |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setFieldOfViewRead Request body: <pre> {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":350.0, "allowSideEffects":true} </pre> |
| Resp. Example | <pre> {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":350.0} </pre> |

2.7.9.2.95 getFieldOfViewPhase

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getFieldOfViewPhase |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" }</pre> |
| Description | Retrieves the field of view dimension in phase direction configured in the protocol for all slice groups (there is only one field of view that applies to all slice groups in the protocol). Unit: in % of field of view read |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getFieldOfViewPhase?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":81.3}</pre> |

2.7.9.2.96 setFieldOfViewPhase

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setFieldOfViewPhase |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Sets the field of view dimension in phase direction for all slice groups in the protocol. |

| | |
|---------------|---|
| | Unit: in % of field of view read |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setFieldOfViewPhase</p> <p>Request body:</p> <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 84.4, "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 84.4 }</pre> |

2.7.9.2.97 getFlipAngle

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getFlipAngle |
| Request Properties | <p>Request Type: GET</p> <p>Required Privilege Level : Execute</p> |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" }</pre> |
| Description | <p>Retrieves the flip angle configured in the protocol.</p> <p>Unit: degree</p> |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getFlipAngle?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</p> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "value": 25.0 }</pre> |

2.7.9.2.98 setFlipAngle

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setFlipAngle |
| Request Properties | <p>Request Type: POST</p> <p>Required Privilege Level : Execute</p> <p>Requires Host Control</p> |
| Parameters | <pre>{ "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": {</pre> |

| | |
|---------------|---|
| | <pre> "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" } </pre> |
| Description | Sets new flip angle value to protocol. Unit: degree |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setFlipAngle Request body: <pre> {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 26.0, "allowSideEffects": true} </pre> |
| Resp. Example | <pre> {"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}, "valueSet": 26.0} </pre> |

2.7.9.2.99 getBaseResolution

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getBaseResolution |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" } </pre> |
| Description | Retrieves the protocol's base resolution. Unit: number of pixels |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getBaseResolution?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre> {"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}, "value": 128} </pre> |

2.7.9.2.100 setBaseResolution

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setBaseResolution |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Int32", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" } |
| Description | Sets the protocol's base resolution. Unit: number of pixels |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setBaseResolution Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 192, "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 192 } |

2.7.9.2.101 getPhaseResolution

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getPhaseResolution |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" } |

| | |
|---------------|--|
| Description | Retrieves the protocol's phase resolution. Unit: % |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getPhaseResolution?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":100.0}</code> |

2.7.9.2.102 setPhaseResolution

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setPhaseResolution |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Double", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Sets the protocol's phase resolution. Unit: % |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setPhaseResolution</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":80.0, "allowSideEffects":true}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":80.0}</code> |

2.7.9.2.103 getConcatenations

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | getConcatenations |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" }</pre> |
| Description | Retrieves the protocol's number of concatenations. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getConcatenations?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":1}</pre> |

2.7.9.2.104 setConcatenations

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setConcatenations |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Int32", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Int32" }</pre> |
| Description | Sets the protocol's number of concatenations. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setConcatenations |

| | |
|------------------|---|
| | Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":2, "allowSideEffects":true} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":2} |

2.7.9.2.105 getBandwidth

| | |
|-----------------------|---|
| Service | parameter/standard |
| Request | getBandwidth |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" } |
| Description | Retrieves the protocol's bandwidth. Unit: Hz / Pixel |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getBandwidth?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":260.0} |

2.7.9.2.106 setBandwidth

| | |
|-----------------------|---|
| Service | parameter/standard |
| Request | setBandwidth |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Double", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered |

| | |
|---------------|---|
| | <pre>clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</pre> |
| Description | Sets the protocol's bandwidth. Unit: Hz / Pixel |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setBandwidth Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":220.0, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":220.0}</pre> |

2.7.9.2.107 getTR

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getTR |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" }</pre> |
| Description | Retrieves the protocol's repetition time. Unit: ms |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTR?sessionI d=26eb060c-553d-4c15-bf5b-23cf29012763</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":100.0}</pre> |

2.7.9.2.108 setTR

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setTR |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Double", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" } |
| Description | Sets new repetition time value to protocol. Unit: ms |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTR Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 400.0, "allowSideEffects": true } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": 400.0 } |

2.7.9.2.109 getTE

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getTE |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" } |

| | |
|---------------|---|
| Description | Retrieves the protocol's echo time. Unit: ms |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getTE?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":10.0}</code> |

2.7.9.2.110 setTE

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | setTE |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <code>{ "value": "Double", "allowSideEffects": "Boolean" }</code> |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Double" }</code> |
| Description | Sets new echo time value to the protocol. Unit: ms |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTE</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":20.0, "allowSideEffects":true}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":20.0}</code> |

2.7.9.2.111 getGoToIsocenter

| | |
|--------------------|---|
| Service | parameter/standard |
| Request | getGoToIsocenter |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | n/a |

| | |
|---------------|--|
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" }</pre> |
| Description | Retrieves the information whether isocenter table positioning mode should be enforced for this step (independent of the global settings). |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/getGoToIsocenter?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":true}</pre> |

2.7.9.2.112 setGoToIsocenter

| | |
|--------------------|--|
| Service | parameter/standard |
| Request | setGoToIsocenter |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Boolean", "allowSideEffects": "Boolean" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Boolean" }</pre> |
| Description | Sets the information whether isocenter table positioning mode should be enforced (independent of the global settings) |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setGoToIsocenter Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":true, "allowSideEffects":true}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "valueSet":true}</pre> |

2.7.10 Table Service

2.7.10.1 Introduction

The table service was newly introduced in version 2 of the Access-i interface to reflect the new / changed table positioning modes that are available for the Numaris X Sola / Vida systems.

In contrast to Access-i version 1, the table positioning mode is set globally for the whole workflow, i.e. not template-/protocol-specific. Therefore a dedicated service was introduced for this purpose.

Three different table positioning modes are available:

- a) **Isocenter**
Before each scan, the table moves to the center of gravity of the slices defined in the protocol to ensure optimal image quality.
- b) **Local Range**
The local range position defines the center of a transversal tolerance range which in most cases has a height of 200 mm. If the protocol slice groups' center of gravity is within this range, the scan will be executed at table position == local range position. If, on the other hand, the protocol slice groups' center of gravity is outside this tolerance range, the table positioning mode behaves like being in Isocenter mode and moves to the center of gravity of the defined slices to ensure good image quality. This mode was introduced for diagnostic scans to have a good tradeoff between minimizing table movements and still having good image quality.
- c) **Fix**
The table position is defined by the user / remote client and does not change until it is set to a different value. If the tech moves the table to home or a different position, the table will move back to the defined fix position when a new scan starts.

2.7.10.2 Table Service Operations

| | |
|---|-----|
| 2.7.10.2.1 getCurrentTablePosition | 165 |
| 2.7.10.2.2 getTablePositioningMode | 166 |
| 2.7.10.2.3 setTablePositioningMode..... | 166 |
| 2.7.10.2.4 getFixTablePosition | 167 |
| 2.7.10.2.5 setFixTablePosition | 167 |
| 2.7.10.2.6 getLocalRangePosition | 168 |
| 2.7.10.2.7 setLocalRangePosition | 168 |
| 2.7.10.2.8 connectServiceToDefaultWebSocket..... | 169 |
| 2.7.10.2.9 connectServiceToWebSocket..... | 169 |
| 2.7.10.2.10 disconnectServiceFromDefaultWebSocket | 170 |
| 2.7.10.2.11 disconnectServiceFromWebSocket | 171 |
| 2.7.10.2.12 resendServiceStateViaWebSockets..... | 171 |

2.7.10.2.1 getCurrentTablePosition

| | |
|--------------------|---|
| Service | table |
| Request | getCurrentTablePosition |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Integer" }</pre> |
| Description | Returns the current table position. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/table/getCurrentTablePosition?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":120}</pre> |

2.7.10.2.2 getTablePositioningMode

| | |
|--------------------|--|
| Service | table |
| Request | getTablePositioningMode |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "isoCenter localRange fix" }</pre> |
| Description | Returns the globally configured table positioning mode that will be used for subsequent measurements. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/table/getTablePositioningMode?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":"fix"}</pre> |

2.7.10.2.3 setTablePositioningMode

| | |
|--------------------|--|
| Service | table |
| Request | setTablePositioningMode |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "isoCenter localRange fix" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl templateRunning noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Set Table Positioning Mode to be used for subsequent measurements. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setTablePositioningMode Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":"isoCenter"}</pre> |

| | |
|---------------|---|
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |
|---------------|---|

2.7.10.2.4 getFixTablePosition

| | |
|--------------------|---|
| Service | table |
| Request | getFixTablePosition |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Integer" }</pre> |
| Description | Returns the currently configured fix table position. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/table/getFixTablePosition?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}, "value":123}</pre> |

2.7.10.2.5 setFixTablePosition

| | |
|--------------------|---|
| Service | table |
| Request | setFixTablePosition |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "value": "Integer" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl templateRunning noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Sets the fix table position that will be applied for subsequent measurements if the table positioning mode is set to "fix". |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setFixTablePosition |

| | |
|------------------|---|
| | Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 123 } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.10.2.6 getLocalRangePosition

| | |
|-----------------------|--|
| Service | table |
| Request | getLocalRangePosition |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Integer" } |
| Description | Returns the currently configured local range position. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/table/getLocalRangePosition?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "value": -230 } |

2.7.10.2.7 setLocalRangePosition

| | |
|-----------------------|--|
| Service | table |
| Request | setLocalRangePosition |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "value": "Integer" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl templateRunning noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |

| | |
|---------------|---|
| Description | Sets the local range position that will be applied for subsequent measurements if the table positioning mode is set to "localRange". |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/standard/setLocalRangePosition</p> <p>Request body:</p> <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": -230 }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

2.7.10.2.8 connectServiceToDefaultWebSocket

| | |
|--------------------|---|
| Service | table |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/table/connectServiceToDefaultWebSocket</p> <p>Request body:</p> <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } }</pre> |

2.7.10.2.9 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | table |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |

| | |
|---------------|---|
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | <p>Enables Websockets for this service</p> <p>Prerequisites:</p> <ul style="list-style-type: none"> - none |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/table/connectServiceToWebSocket</p> <p>Request body:</p> <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.10.2.10 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|---|
| Service | table |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | <p>Request Type: POST</p> <p>Required Privilege Level : Read</p> |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | <p>Disables Websockets for this service</p> <p>Prerequisites:</p> <ul style="list-style-type: none"> - none |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/table/disconnectServiceFromDefaultWebSocket</p> <p>Request body:</p> <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.10.2.11 disconnectServiceFromWebSocket

| | |
|--------------------|--|
| Service | table |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/table/disconnectServiceFromWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.10.2.12 resendServiceStateViaWebSockets

| | |
|--------------------|--|
| Service | table |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |

| | |
|---------------|---|
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/table/resendServiceStateViaWebSockets</pre> <p>Request body:</p> <pre>{"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}}</pre> |

2.7.10.3 Tips and Common Pitfalls

To ensure the intended table behavior of the remote client software, some precautionary measures have to be taken:

Scenario 1: Fix Table Position

Before start of each scan, make sure that

- a) table positioning mode is set to fix and the fix table position is set to the desired value
- b) parameter/standard/getGoTolsocenter should be false, if not, call setGoTolsocenter(false). This is important as the step-specific "GoTolsocenter" protocol parameter overrules the globally selected table mode.

Scenario 2: Table should not move automatically ("Table Lock")

- a) Set table mode to fix.
- b) Before each scan
 - a. make sure that parameter/standard/getGoTolsocenter returns value == false.
 - b. read current table position and set it as fix table position.

This guarantees that the table will not move when the scan starts (nonetheless, if the slice positions are too far offcenter, the measurement may fail).

2.7.11 Image Service

2.7.11.1 Introduction

Via the websocket functionality the client can automatically receive images that are reconstructed on the MR host.

2.7.11.2 Images in MRI

Slice

A Slice is an MR image given in the Patient Coordinate System, which has to be scanned or is already scanned.

Plane of Slice/Image

This is a simplified (mathematical) model of a Slice. The slice thickness and the extent of the FoV (see below) are neglected. The Plane of Slice/Image is treated as if it were infinite. The image plane is important for other definitions like "Slice Shift" and "Slice Offcenter" and for mathematical calculations.

Field of View

The Field of View (FoV) FOV is the square image area that contains the object of interest to be measured. The two sides of a FoV are called Readout and Phase Encoding direction. The size of a FoV is given in both directions in mm.

Image Position in DICOM

Image Position (Patient) (0020,0032). This points to the center of the image's left upper voxel. The definition of Image Position in DICOM and in *syngo* MR are shown in Figure 16.

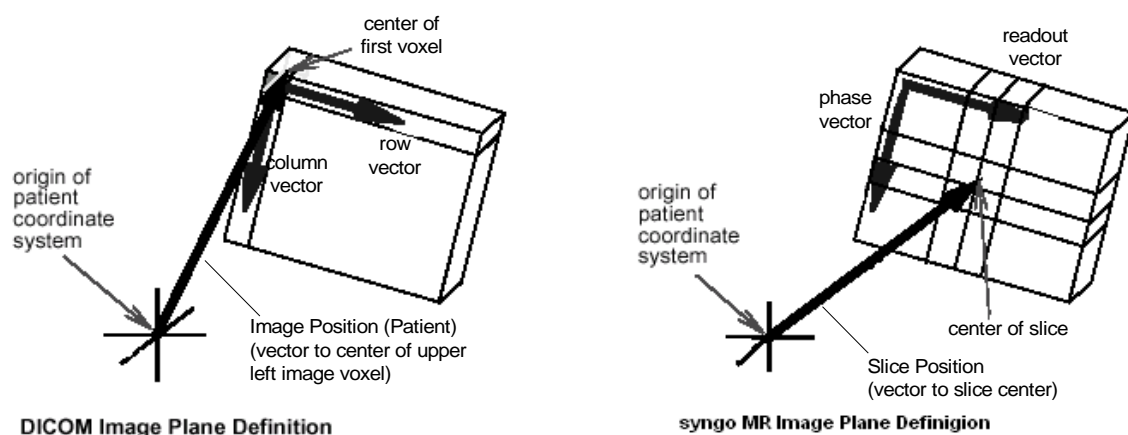


Figure 16: Image Position in DICOM and syngo MR.

Slice Center

The Slice Center specifies the exact center position of a slice (of the FoV). The Position Vector of a Slice points to the Slice Center. The coordinates of the Slice Center are given in the Patient Coordinate System.

Attention: The Slice Center differs from the Image Position (Patient).

Frame of Reference

The Frame of Reference Module is defined in DICOM and insures the spatial relationship of images within a series. The Frame of Reference becomes invalid by following events:

- Registration of a patient
- Positioning of a patient via the light-beam localizer
- Movement of the table board into home position
- Manual movement of the table board
- Alternation between "Scanner ready" <-> "Scanner not ready"

2.7.11.3 Coordinate Systems

2.7.11.3.1 DICOM Image Patient Coordinate System (dicomPcs)

The DICOM Image Patient Coordinate System (dicomPcs) is a two dimensional coordinate system. It represents the space occupied by an image and does so in 2D image pixel units. It is defined by three vectors:

- DICOM Image Position (Patient)
- DICOM Image Orientation (Patient) → Row, Column

The Image Position (Patient) points from the origin of the Patient Coordinate System to the center of the left upper pixel of the given image.

The origin of the dicomPcs is defined by the center of the left upper pixel of the image.

The Row vector and the Column vector define the plane in which the image lies. Their directions are specified in the DICOM attribute Image Orientation (Patient). They originate in the point, which the Position vector points to. Row and Column have to be orthogonal. The x axis increases positively in the right direction, and the y axis increases positively towards the bottom.

Important for some calculations is also the Normal Vector, which is the cross-product of Column and Row (Column \times Row).

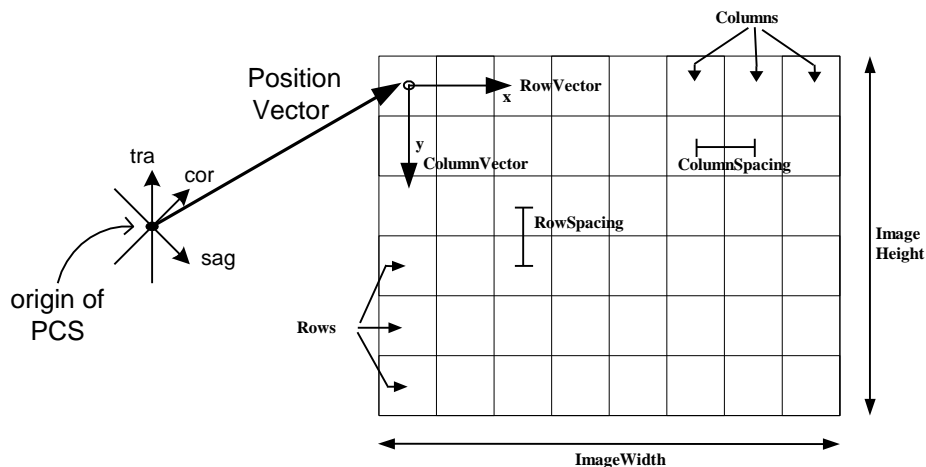


Figure 17: Illustration of dicom patient coordinate system.

2.7.11.3.2 MR Slice Patient Coordinate System (mrSlicePcs)

Axes, Origin and Planes

The syngo MR Patient Coordinate System (mrSlicePcs) is derived from the DICOM coordinate system. It uses the same axes. The sagittal axis increases to the left, the coronal axis to the posterior and the transverse axis to the head (see Figure 18). The measurements are given in mm.

Remember that the *syngo* MR notation of the axes is different from the notation in medical science.

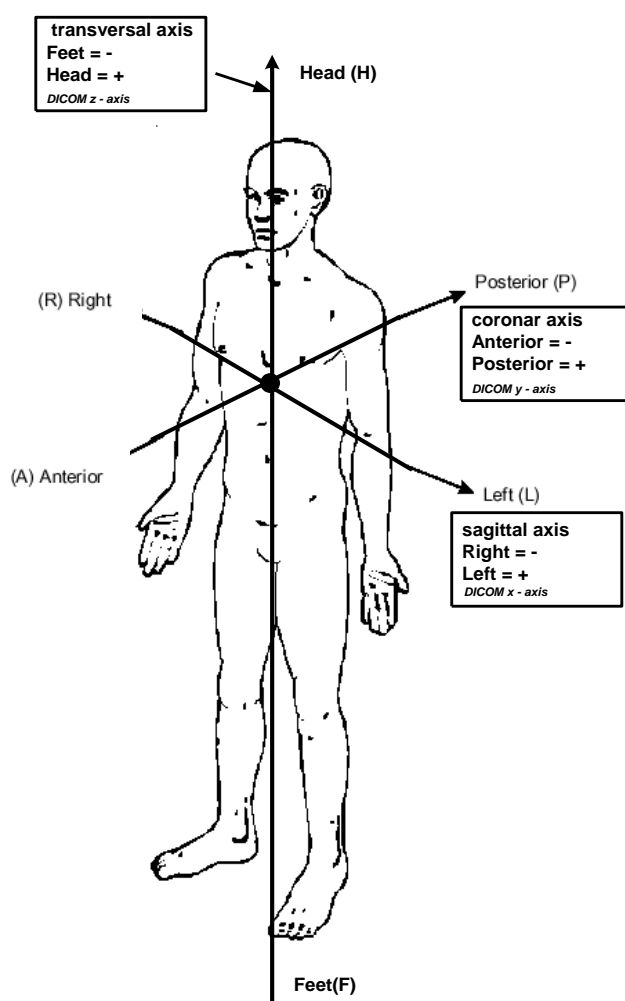


Figure 18: The Patient Coordinate System

The origin of the coordinate system is given by the ISO center of the magnet. The ISO Center is the origin of the gradient system. Before moving the patient table into the magnet, a position on the table is marked by a light marker, and this position is then moved into the ISO center of the magnet. All subsequent images, for which the origin of the coordinate system has been set in that way, have the same DICOM Frame of Reference UID if the table is no more moved.

The orientation of the PCS depends solely on the patient registration and is independent from the patient's orientation on the table.

A point is exactly specified by giving the coordinates in sagittal (right → left), coronal (anterior → posterior) and transverse (feet → head) direction.

The image planes are the same than defined in DICOM. There is a convention for loading images into the viewer (see Figure 19):

Sagittal images are presented with the anterior side showing to the left of the observer.

Coronal images are presented with the anterior side towards the observer.

Transverse images are presented cranial.

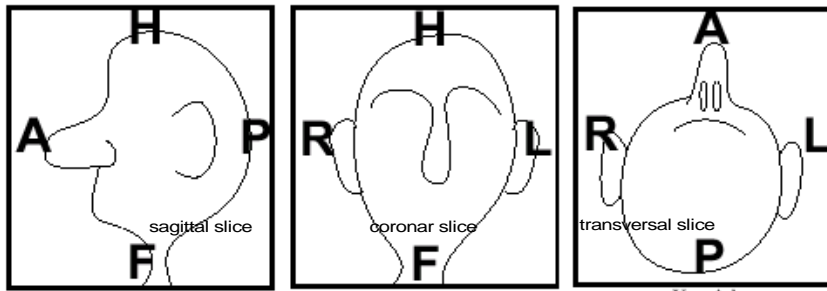


Figure 19: Default Presentation of Slices in Viewer

Important Image Attributes

The position, orientation and size of an image is specified by the DICOM attributes.

Image Position (Patient) points to the center of the left upper voxel of the image.

The row and column vectors are described by their coordinates in sag, cor and tra direction. In DICOM they are specified in the attribute Image Orientation (Patient).

Figure 20 shows the definition of Slice Shift (In the image text "SP"). This is the same than Slice Location in DICOM and means the distance from the ISO –Center to the image plane in the direction of the normal vector.

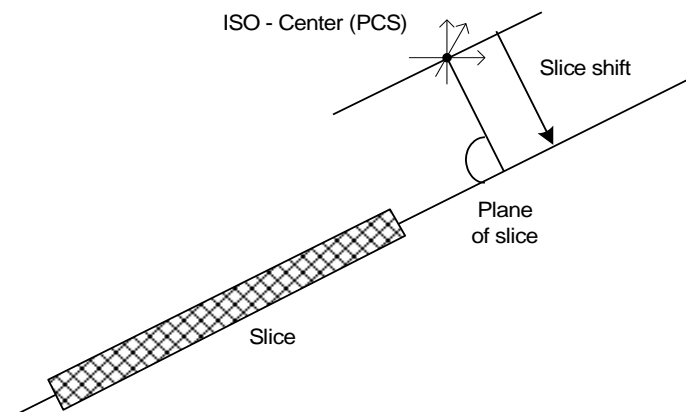


Figure 20: Illustration of Slice Shift (=Slice Location,=Slice Position = SP)

Another important item is the Slice Offcenter (see Figure 21). This is the distance of the projection of the ISO-Center to the slice center.

A slice has one Slice OffCenter in Readout and one in Phase direction.

Slice Shift and Slice OffCenter describe the exact position of a slice in the Patient Coordinate System.

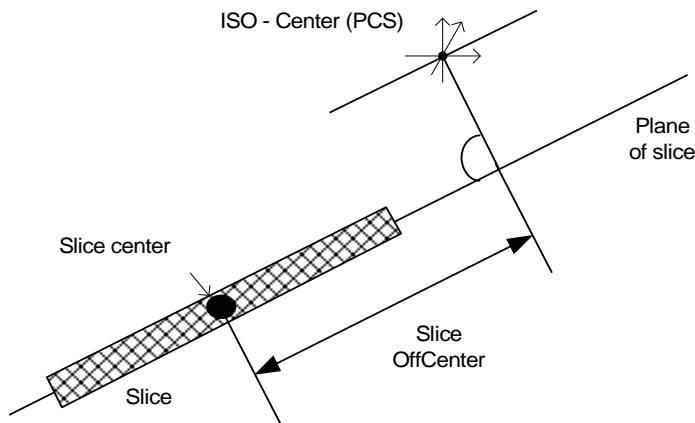


Figure 21: Illustration of Slice OffCenter

Transformations

Transformation from dicomPcs to mrSlicePcs: Figure 22 shows the relation between MR Slice Patient Coordinate System and DICOM Image Patient Coordinate System.

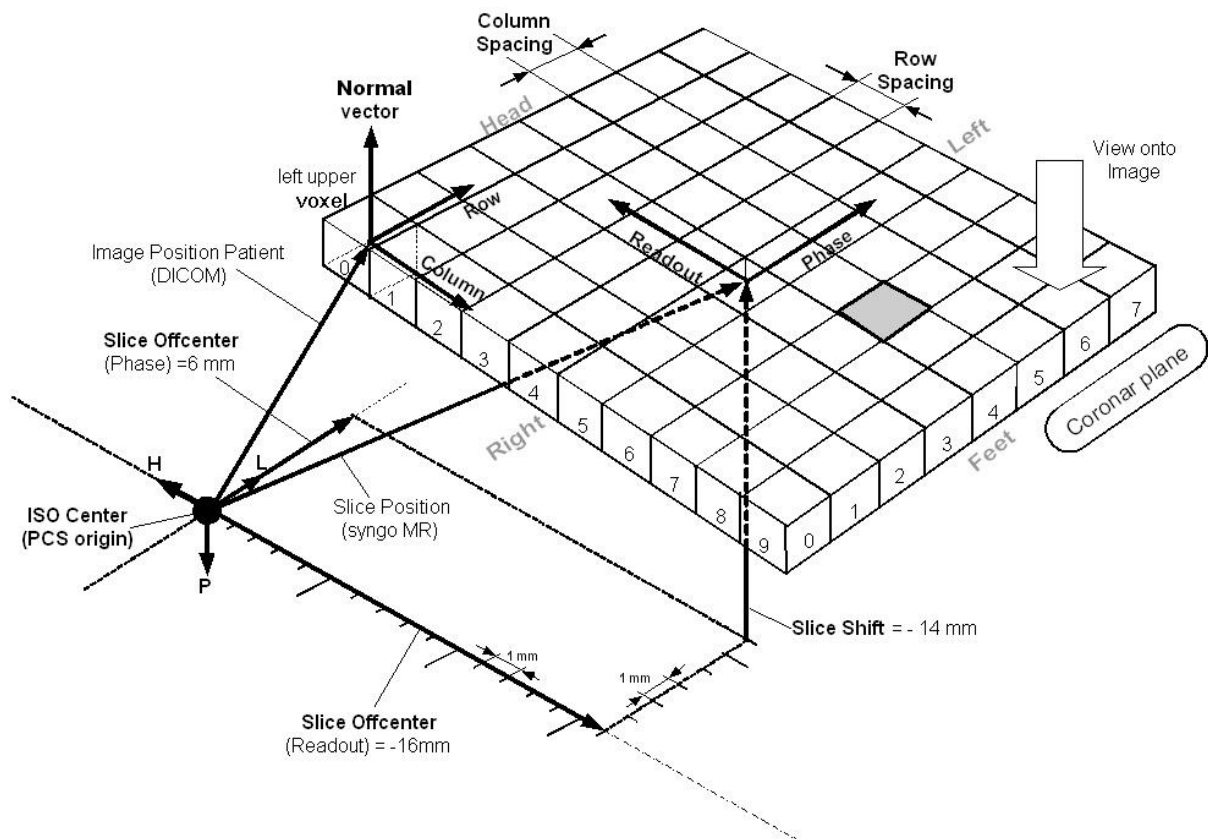


Figure 22: The Relation of MR Slice Patient Coordinate System to DICOM Image Patient Coordinate System

In the drawn example a coronal plane has been taken. A pixel is 2 mm in height and width. So the pixel spacing in row direction and in column direction is 2 mm, and 1 pixel contains 4 mm². dicomPcs origin described by the DICOM Image Position (Patient) (0020,0032) and pointing to the center of the upper left voxel is (-1, -14, -7) (3 ½ pixel = 7 mm right and 4 ½ pixel = 9 mm head of the Image Position syngo MR).

The row vector corresponds to the Right \Leftrightarrow Left axis (x axis) and increases towards the left side of the patient, so it has the coordinates (1,0,0). The column vector corresponds to the Head \Leftrightarrow Feet axis (z axis) and increases towards the feet, so it has the coordinates (0,0,-1).

In the lower left corner is the origin of the mrSlicePcs.

Slice Offcenter (Phase) is 6 mm to left \rightarrow sag coordinate = 6 mm

Slice Offcenter (Readout) is 16 mm to the feet \rightarrow tra coordinate = -16 mm

Slice Shift is 14 mm upwards (anterior) \rightarrow cor coordinate = -14 mm

Slice Position (pointing to the center of the image) is L6 A14 F16, meaning 6 mm to left, 14 mm to anterior, 16 mm to the feet, seen from the origin of the mrSlicePcs pointing to the center of the image.

FoV: The image measures 20 mm in readout direction and 16 mm in phase direction, so the FoV is 20 x 16 mm.

Note: There is an in-plane half pixel deviation between the planned and resulting image center. For further details please refer to Ref. [2].

2.7.11.3.3 MR Slice Device Coordinate System (mrSliceDcs)

The MR Device Coordinate System (mrSliceDcs) is a three-dimensional coordinate system which is defined by the hardware and the magnetic field. The origin of the mrSliceDcs is in the ISO center of the magnetic field. The x axis, the y axis and the z axis form a right handed coordinate system. The direction of the axes depends on the hardware and correlates to the x-, y- and z-axis of the gradient system. The measurements are given in mm.

Figure 23 shows the axes of the default scanner model. The thin arrow shows the origin of the mrSliceDcs is identical to the ISO-Center of the magnetic field and the origin of the gradient system.

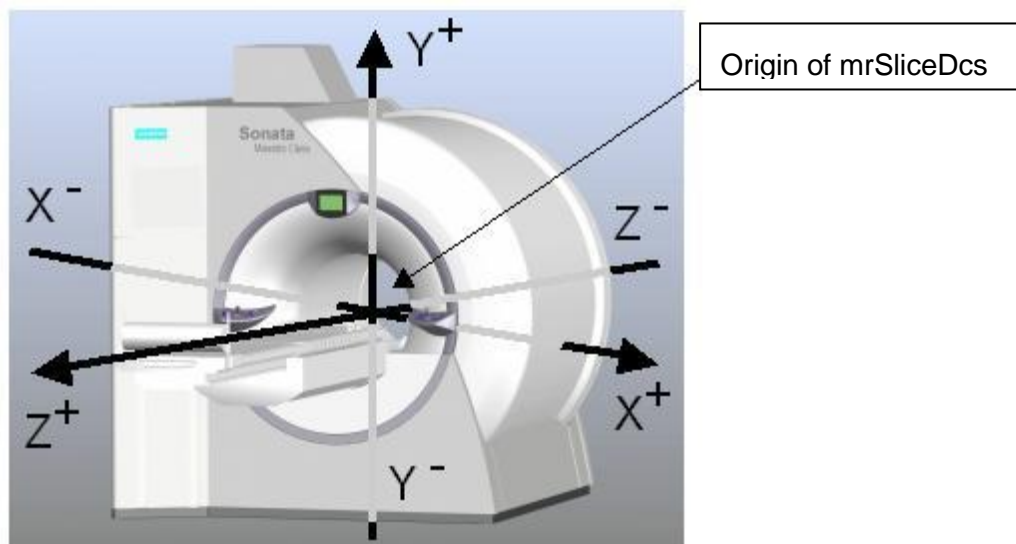


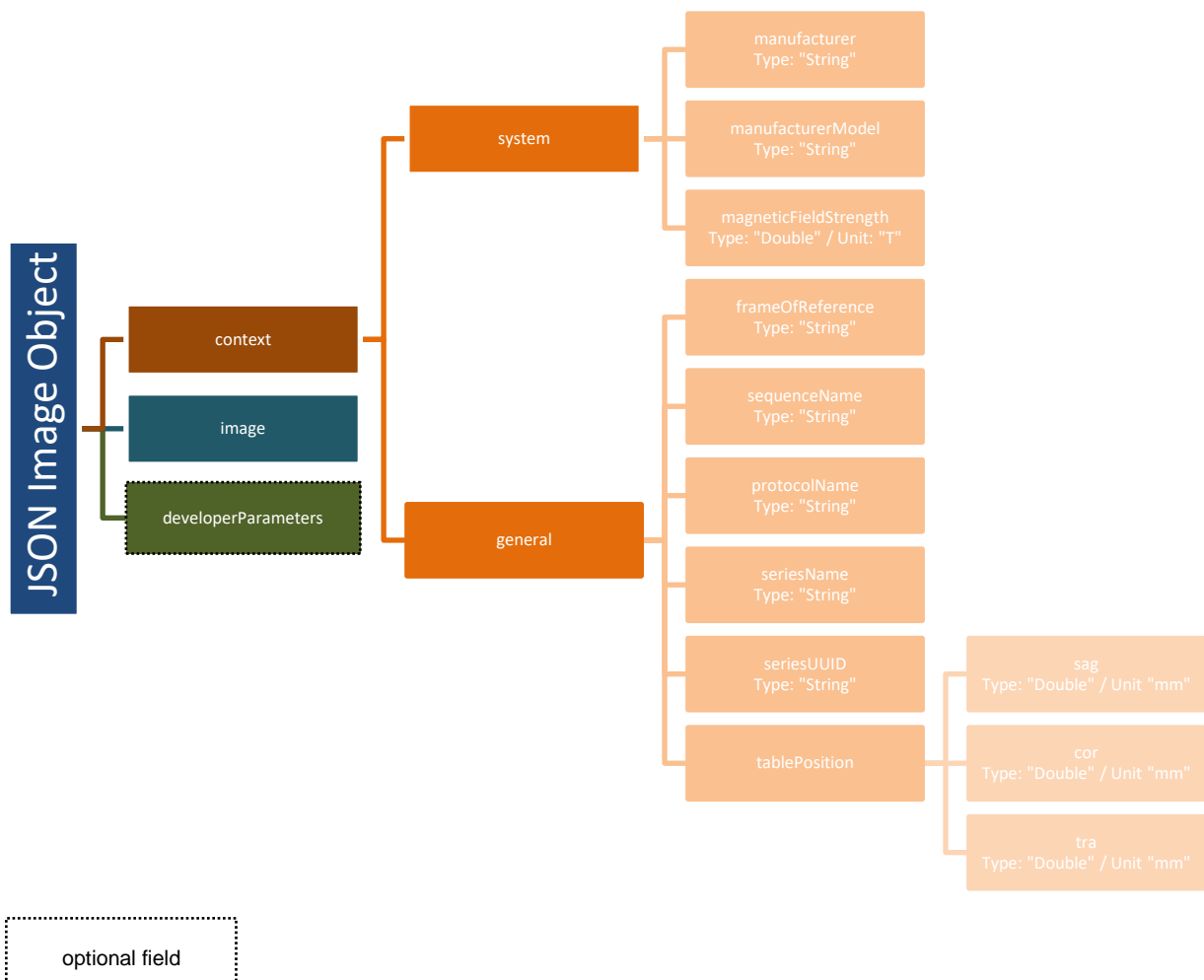
Figure 23: The mrSliceDcs in Symphony/Harmony

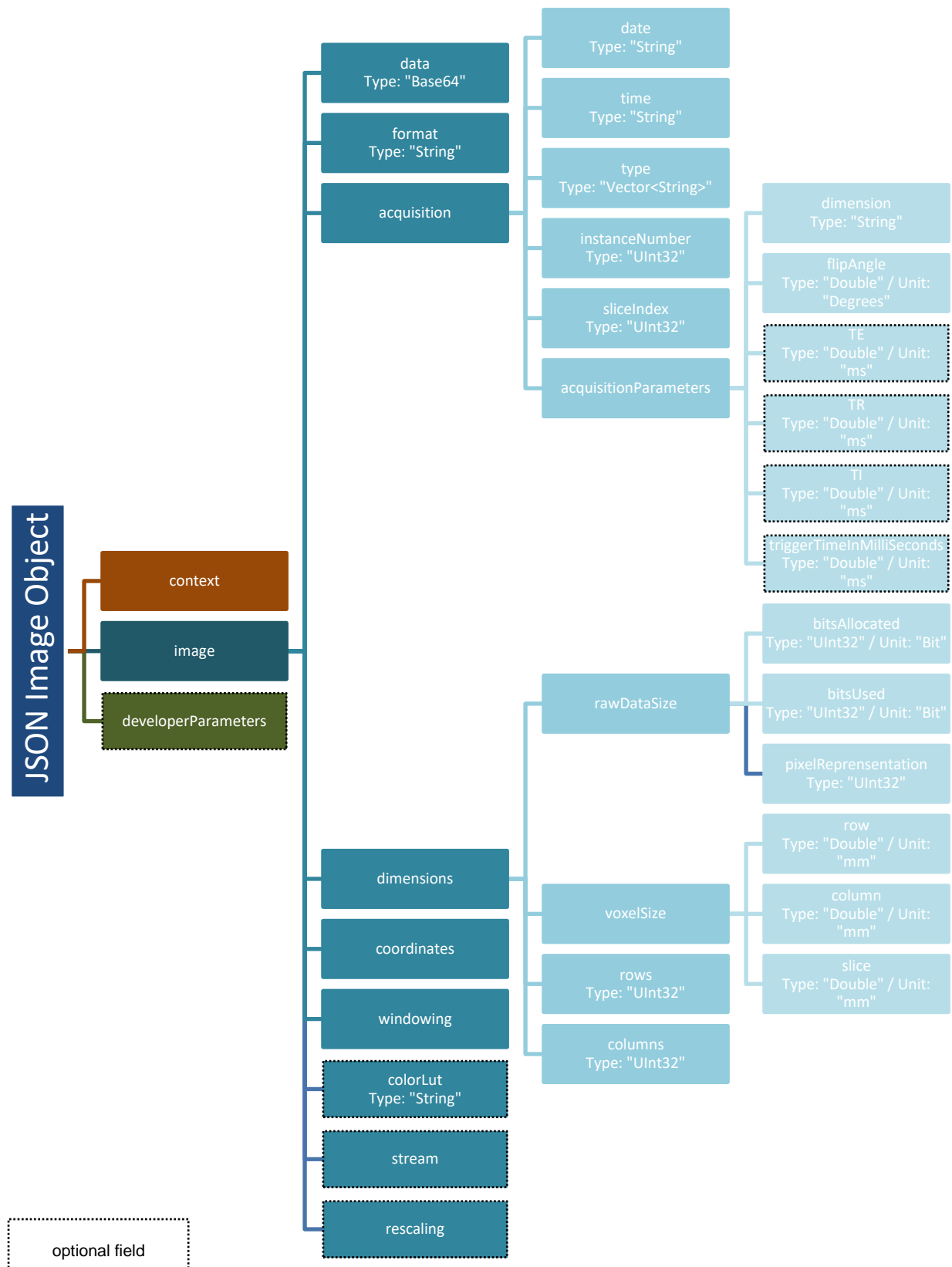
2.7.11.3.4 DICOM Image Device Coordinate System (dicomDcs)

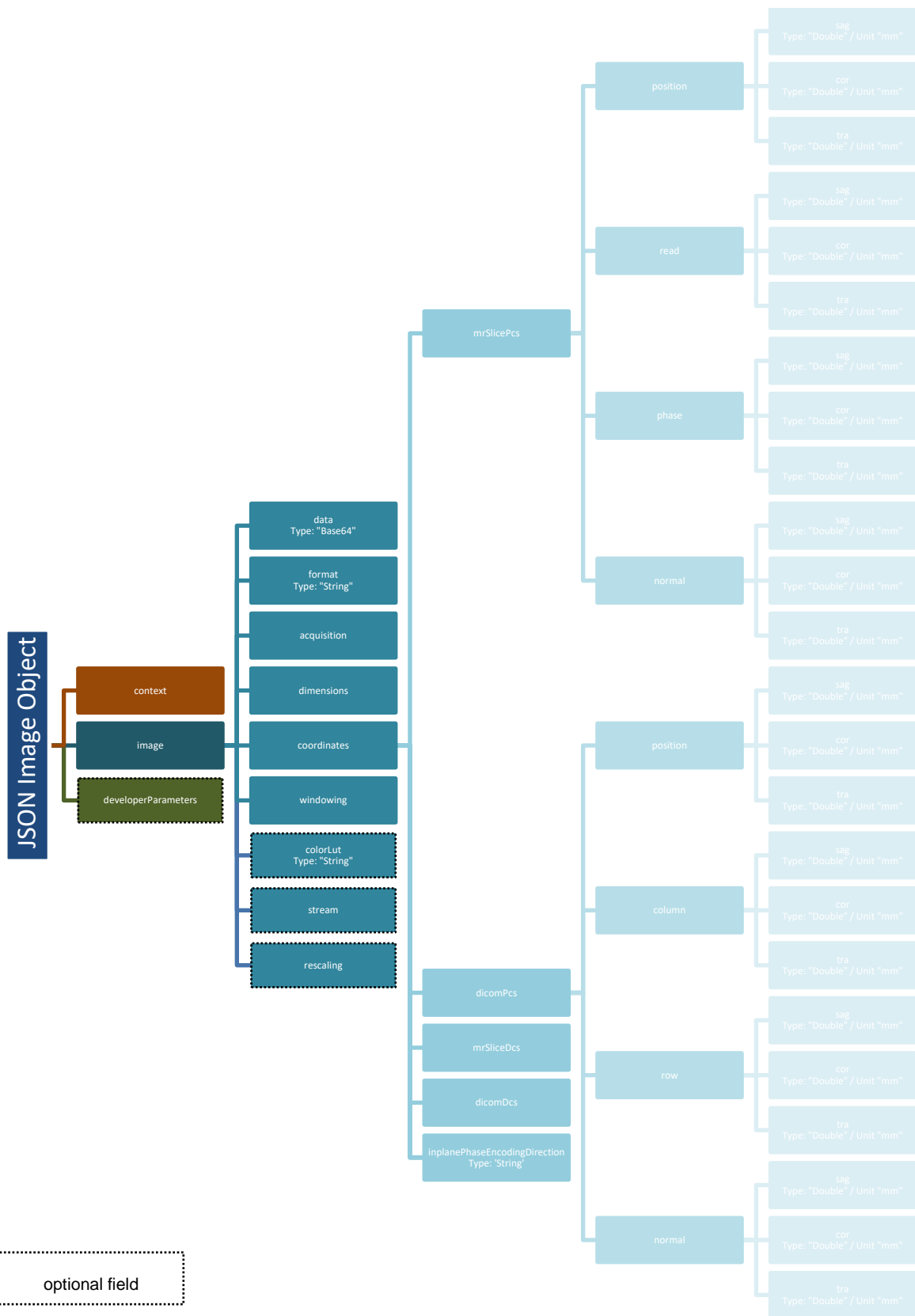
The DICOM Device Coordinate System (dicomDcs) defines the coordinate space similar as the regular dicomPcs but with respect to the ISO center of the magnetic field.

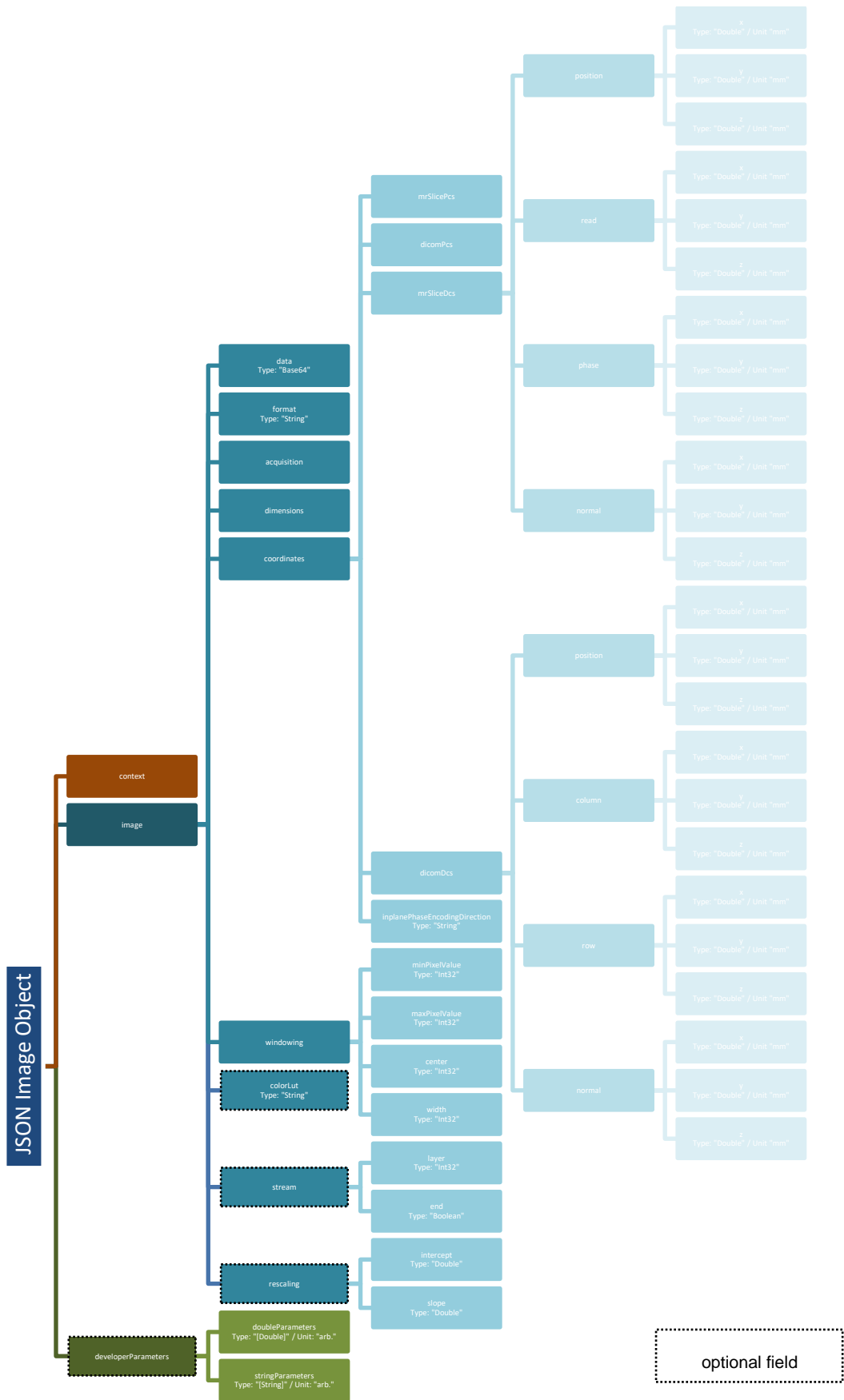
2.7.11.4 Image Format

The JSON encoded image struct aims to provide all necessary information for real-time image processing, i.e. all spatial information to process and selected acquisition properties to uniquely identify the image in a comfortable manner. The actual image is encoded within the `[image][data]` field of the *websocket response message*. The RC can choose either between the *raw16bit* or *dicom* formatting of the image. The information is filled and created directly after image reconstruction by the *ScannerRemoteControlSender*.









2.7.11.4.1 Raw 16bit

The raw 16 byte image data is base64 encoded and can be accessed in the image data field. Please note that the byte stream is filled line by line. An image can be obtained by using the following pseudo code:

```
imageDataBase64 = JsonImgObj[image][data]
imageDataByte = base64ToByte(imageDataBase64)
image[JsonImgObj[image][dimensions][rows], JsonImgObj[image][dimensions][columns]]
    = typecast(imageDataByte, 'uint16')
```

2.7.11.4.2 DICOM

Similar to *raw16bit* format, the DICOM image can be obtained from byte array encoded in the image data field:

```
dicomByteArray = base64ToByte(JsonImgObj[image][data])
dicomImg = MyDicomClass::loadFromByteArray(dicomByteArray)
```

Please note that the DICOMs provided by the SRC image fast track are not equal with those of in the database. More precisely, only a subset of DICOM tags is supported (Table 1).

Table 1: Dicom Tags supported by fast track.

| Tag | Description | Tag | Description |
|------------|-------------------------|------------|--------------------------|
| '0008,0008 | Image Type | '0018,1312 | Phase encoding direction |
| '0008,0022 | Acquisition date | '0018,1314 | Flip angle |
| '0008,0032 | Acquisition time | '0018,5100 | Patient position |
| '0008,103E | Series description | '0020,000D | Study UID |
| '0010,0010 | Patient name | '0020,0010 | Study number |
| '0010,0020 | Patient ID | 0020,000E | Series Instance UID |
| '0018,0050 | Slice thickness | '0020,0013 | Image number |
| '0018,0080 | Repetition time | '0020,0032 | Image position |
| '0018,0081 | Echo time | '0020,0037 | Image orientation |
| '0018,0083 | Number of Averages | '0020,1041 | Slice location |
| '0018,0084 | Imaging frequency | '0028,0010 | Rows number |
| '0018,0087 | Magnetic field strength | '0028,0011 | Columns number |
| '0018,0095 | Pixel bandwidth | '0028,0030 | Pixel spacing |
| '0018,1310 | Acquisition matrix | '0028,0100 | Bits allocated |

2.7.11.5 Image Service Operations

| | |
|--|-----|
| 2.7.11.5.1 setImageFormat | 186 |
| 2.7.11.5.2 getLastSeriesNumber | 186 |
| 2.7.11.5.3 connectServiceToDefaultWebSocket | 187 |
| 2.7.11.5.4 connectServiceToWebSocket | 187 |
| 2.7.11.5.5 disconnectServiceFromDefaultWebSocket | 188 |
| 2.7.11.5.6 disconnectServiceFromWebSocket | 188 |
| 2.7.11.5.7 resendServiceStateViaWebSockets | 189 |

2.7.11.5.1 setImageFormat

| | |
|--------------------|---|
| Service | image |
| Request | setImageFormat |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "value": "dicom raw16bit" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError unsupportedImageFormat requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Defines the image format that a client wants to receive via the Image Service. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/image/setImageFormat Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": "dicom" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.11.5.2 getLastSeriesNumber

| | |
|--------------------|--|
| Service | image |
| Request | getLastSeriesNumber |
| Request Properties | Request Type: GET Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Int32" } |
| Description | Get the series number of the last measured series |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/image/getLastSeriesNumber?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Examp. | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "value": 1 } |

2.7.11.5.3 connectServiceToDefaultWebSocket

| | |
|--------------------|---|
| Service | image |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/image/connectServiceToDefaultWebSocket Request body: {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Exam. | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.11.5.4 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | image |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/image/connectServiceToWebSocket Request body: |

| | |
|---------------|--|
| | <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.11.5.5 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|---|
| Service | image |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/image/disconnectServiceFromDefaultWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.11.5.6 disconnectServiceFromWebSocket

| | |
|--------------------|---|
| Service | image |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |

| | |
|---------------|--|
| | <pre>} }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/image/disconnectServiceFromWebSoc ket</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.11.5.7 resendServiceStateViaWebSockets

| | |
|--------------------|---|
| Service | image |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/image/resendServiceStateViaWebSoc kets</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.11.6 Websocket Messages

Regarding images, a client retrieves 3 kinds of websocket packages:

- **ImageStreamStart**
Signals the beginning of an image stream and contains the following information about the protocol template step that produces the images:
 - id: template-id of the protocol template step
 - label: the step's label
 - isInteractive: tells whether the protocol is interactive
- **ImageStream**
Contains the image meta data and the image itself (in Base64 encoding)
- **ImageStreamEnd**
Signals the end of an image stream.

The order of package retrieval is as follows:

```
imageStreamStart
Loop
{
    imageStream
}
imageStreamEnd
```

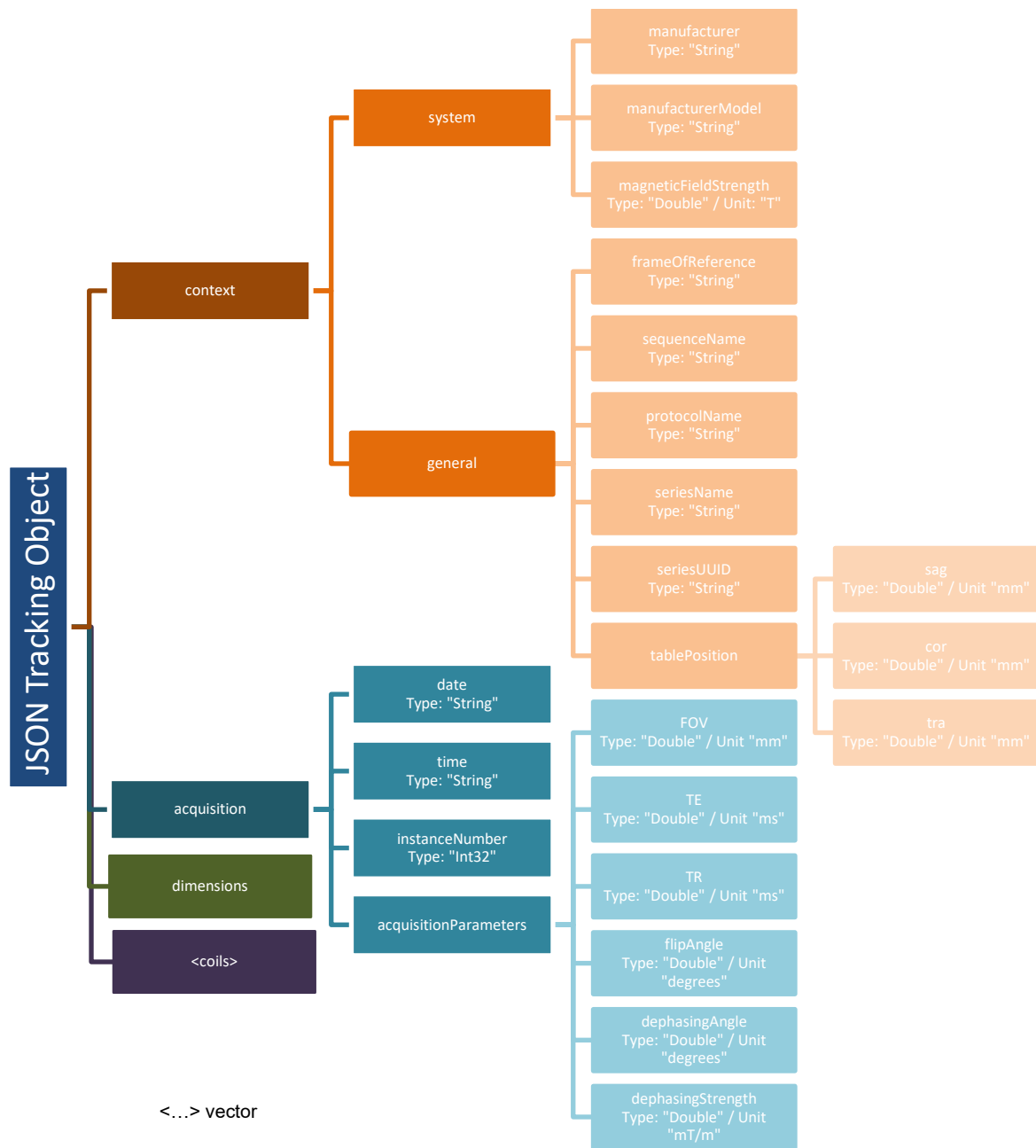
2.7.12 Tracking Service

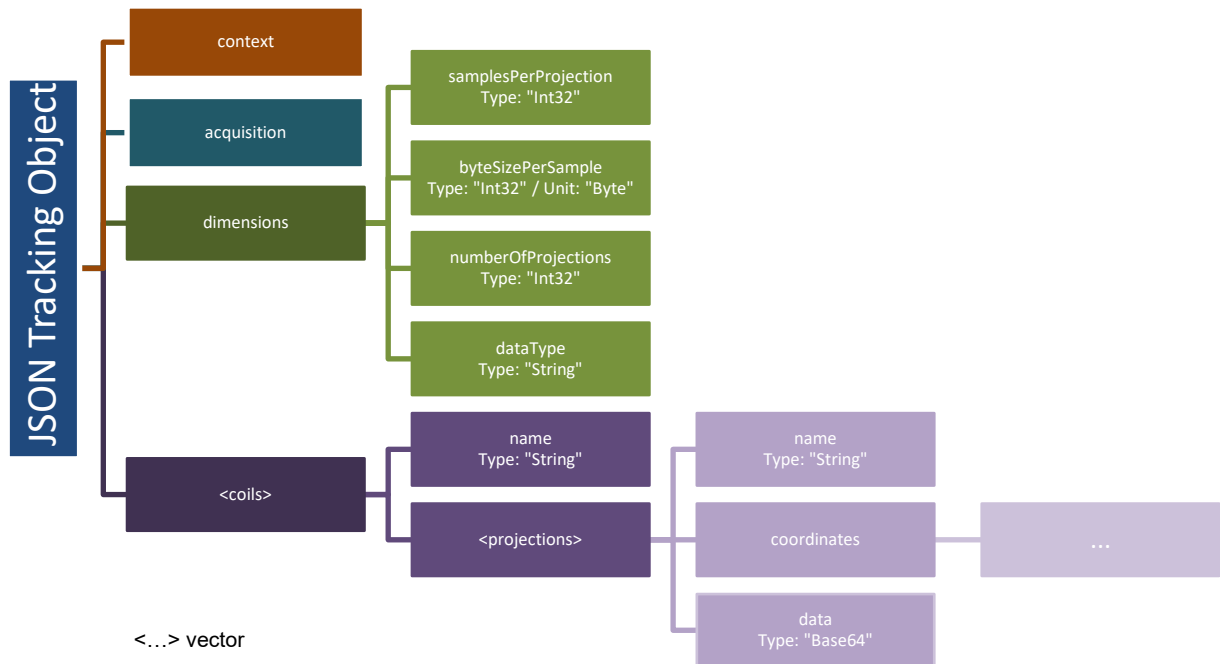
2.7.12.1 Introduction

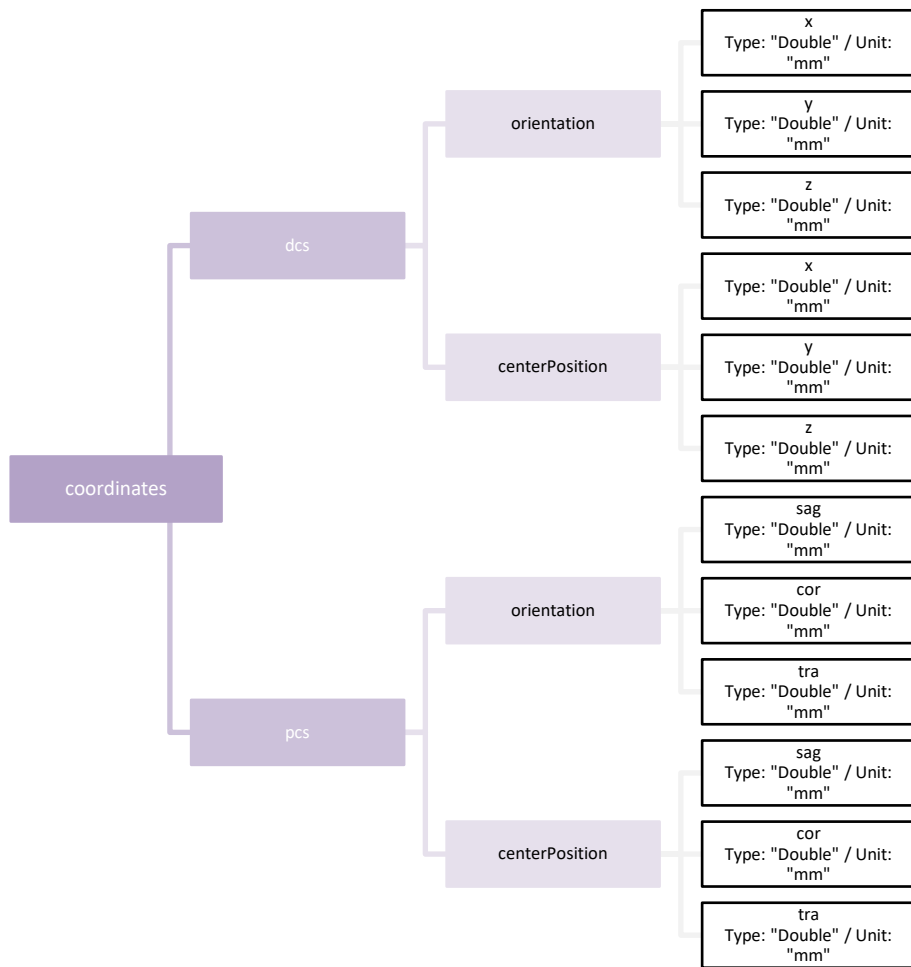
If supported by a projection sequence running on the host, position projections of (micro) MR receiver coils are automatically sent to the client via websockets.

2.7.12.2 Tracking Format

The JSON encoded tracking struct aims to provide all necessary information for real-time calculation of the microcoil positions, i.e. all spatial information to process and selected acquisition properties to uniquely identify the projections in a comfortable manor. The actual projection data is encoded within the *[coils()][projections()][data]* field of the websocket response message and is provided by the Scanner Remote Control Sender.







The projection data is base64 encoded and can be accessed in the projection data field. Please note that the byte stream can support different dataTypes. A projection can be obtained using the following pseudo code:

```
// First projection data of the first coil
projectionDataBase64 = JsonTrackingObj[coils][0][projections][0][data]
projectionDataByte = base64ToByte(imageDataBase64)
projection[JsonTrackingObj[dimensions][samplesPerProjection]] =
    typecast(imageDataByte, JsonTrackingObj[dimensions][dataType])
```

2.7.12.3 Tracking Service Operations

| | |
|---|-----|
| 2.7.12.3.1 connectServiceToDefaultWebSocket | 195 |
| 2.7.12.3.2 connectServiceToWebSocket | 195 |
| 2.7.12.3.3 disconnectServiceFromDefaultWebSocket..... | 196 |
| 2.7.12.3.4 disconnectServiceFromWebSocket..... | 196 |
| 2.7.12.3.5 resendServiceStateViaWebSockets | 197 |

2.7.12.3.1 connectServiceToDefaultWebSocket

| | |
|--------------------|---|
| Service | tracking |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/tracking/connectServiceToDefaultWebSocket Request body: {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.12.3.2 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | tracking |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/tracking/connectServiceToWebSocket |

| | |
|------------------|---|
| | Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.12.3.3 disconnectServiceFromDefaultWebSocket

| | |
|-----------------------|--|
| Service | tracking |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/tracking/disconnectServiceFromDefaultWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.12.3.4 disconnectServiceFromWebSocket

| | |
|-----------------------|---|
| Service | tracking |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered |

| | |
|---------------|---|
| | <pre>clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/tracking/disconnectServiceFromWeb Socket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.12.3.5 resendServiceStateViaWebSockets

| | |
|--------------------|---|
| Service | tracking |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/tracking/resendServiceStateViaWeb Sockets Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.12.4 WebSocket Messages

The RC retrieves the current tracking data by WebSocket messages that signal start of tracking data, actual tracking data packages and end of tracking data.

The order of package retrieval is as follows:

```
trackingDataStreamStart
```

```
Loop
```

```
{
```

```
    trackingDataStream
```

```
}
```

```
trackingDataStreamEnd
```

2.7.13 Issue Service

2.7.13.1 Introduction

While working with the scanner a number of different issues may come up that require the attention of the user.

These issues may either just inform the user or require the user to confirm or to decline a specific action that will take place at the scanner.

Access-i supports 9 different issues that can be subdivided into 2 types:

- Simple issues that can only be confirmed or declined without the need of any additional parameters
- Complex issues that need additional information passed in as "parameters" when they are confirmed.

A complete list of the supported issues can be found in Ref. [3].

The "simple" issues are:

- Automatic Table Move (`automaticTableMove`)
- Patient Not Positioned (`patientNotPositioned`)
- Sed Warning (`sedWarning`)
- Stimulation Warning (`stimulationWarning`)
- System Frequency Manipulated (`systemFrequencyManipulated`)
- General (`general`, an issue that should never occur in a remote scenario, can only be declined)

The "complex" issues (that need additional parameters provided to the `handleIssue` call) are

- Patient Not Registered (`patientNotRegistered`)
- SAR Limits Exceeded (`sarLimitsExceeded`)
- Stimulation Limits Exceeded (`stimulationLimitsExceeded`)

It is mandatory for a RC to answer these issues raised via the Issue Solver Service as no protocol may be opened or started while an issue is pending. The type of issue that is raised is transparent to the remote client as all types of issues are accessed via the same generic interface methods. The RC shall either poll `getIssue` regularly via HTTP or receive `getIssue` messages via WebSocket to ensure that pending issues are discovered. If the 'isApplicable' argument within 'getIssue' returns true, the RC should evaluate the handling options within the value argument.

Finally, the RC shall manage the issue via the interface method 'handleIssue' and the appropriate handling option.

NOTE: If the connection to the host gets lost or the host enforces the host control back, the issue gets automatically declined.

2.7.13.2 Complex Issue Parameters

Here the structure of the parameters that need to be provided to the `handleIssue` call of complex issues are described in detail.

Example: sarLimitsExceeded

When a `sarLimitsExceeded` issue is signalled, the following data structure is provided with the `getIssue` response / the `getIssue` WebSocket message:

```
"value": {
  issueType: "sarLimitsExceeded",
  handlingOptions": [
    {
      "id": "confirm",
      "parameters": [
        [
          {
            "key": "proposal",
            "value": "changeTr"
          },
          {
            "key": "value",
            "value": 3815
          }
        ],
        [
          {
            "key": "proposal",
            "value": "changeSlices"
          },
          {
            "key": "value",
            "value": 5
          }
        ],
        ....
      ]
    },
    {
      "id": "decline",
      "parameters": null
    }
  ]
}
```

In this scenario, if the issue is supposed to be confirmed, the remote client may decide whether to change TR or to change slices.

The provided values 3815 or 5 are recommended values, the remote client may decide to take pass different ones to the `handleIssue` call.

The decision, including the chosen value, must be passed as parameter to the `handleIssue` method.

Example:

```
{
  "id": "confirm",
  "parameters": [
    {
      "key": "proposal",
      "value": "changeSlices"
    },
    {
      "key": "value",
      "value": 5
    }
  ],
  "sessionId": "1016d210-f86d-4328-9489-90f0fc4c919b"
}
```

As the "parameters" property of the "decline" handling options is null, no parameters are passed to the `handleIssue` call if the issue is declined.

2.7.13.2.1 Patient Not Registered Issue Parameters

```
"parameters": [
  [{
    "key": "patientMoved",
    "value": false
  }],
  [{
    "key": "patientMoved",
    "value": true
  }]
]
```

2.7.13.2.2 SAR Limits Exceeded Issue Parameters

```
"parameters": [
  [
    {
      "key": "proposal",
      "value": "changeTr"
    },
    {
      "key": "value",
      "value": DOUBLE
    }
  ],
  [
    {
      "key": "proposal",
      "value": "changeSlices"
    },
    {
      "key": "value",
      "value": INT
    }
  ]
]
```

```
    ],  
    [  
      {  
        "key": "proposal",  
        "value": "changeFlipAngle"  
      },  
      {  
        "key": "value",  
        "value": DOUBLE  
      }  
    ]  
  ]  
]
```

2.7.13.2.3 StimulationLimitsExceeded

```
"parameters": [  
  [  
    {  
      "key": "proposal",  
      "value": "changeFov"  
    },  
    {  
      "key": "value",  
      "value": DOUBLE  
    }  
  ],  
  [  
    {  
      "key": "proposal",  
      "value": "changeGradientRiseTime"  
    },  
    {  
      "key": "value",  
      "value": DOUBLE  
    }  
  ],  
  [  
    {  
      "key": "proposal",  
      "value": "changeSliceThickness"  
    },  
    {  
      "key": "value",  
      "value": DOUBLE  
    }  
  ]  
]
```

2.7.13.3 Issue Service Operations

| | |
|--|-----|
| 2.7.13.3.1 connectServiceToDefaultWebSocket..... | 203 |
| 2.7.13.3.2 connectServiceToWebSocket..... | 203 |
| 2.7.13.3.3 disconnectServiceFromDefaultWebSocket | 204 |
| 2.7.13.3.4 disconnectServiceFromWebSocket | 205 |
| 2.7.13.3.5 resendServiceStateViaWebSockets | 205 |
| 2.7.13.3.6 getIssue | 207 |
| 2.7.13.3.7 handleIssue | 208 |

2.7.13.3.1 connectServiceToDefaultWebSocket

| | |
|--------------------|--|
| Service | issue |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/issue/connectServiceToDefaultWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.13.3.2 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | issue |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |

| | |
|---------------|---|
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/issue/connectServiceToWebSocket</pre> Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.13.3 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|---|
| Service | issue |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/issue/disconnectServiceFromDefaultWebSocket</pre> Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.13.3.4 disconnectServiceFromWebSocket

| | |
|--------------------|--|
| Service | issue |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/issue/disconnectServiceFromWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.13.3.5 resendServiceStateViaWebSockets

| | |
|--------------------|--|
| Service | issue |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |

| | |
|---------------|---|
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/issue/resendServiceStateViaWebSockets</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.13.3.6 getIssue

| | |
|--------------------|---|
| Service | issue |
| Request | getIssue |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets Requires Host Control |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "isApplicable": "Boolean", "value": { "issueType": "automaticTableMove patientNotPositioned patientNotRegistered sarLimitsExceeded sedWarning stimulationLimitsExceeded stimulationWarning systemFrequencyManipulated general" }, "handlingOptions": [{ "id": "String", "parameterRange": [{ "key": "String", "value": "Object" }] }], "additionalInformation": [{ "key": "String", "value": "Object" }] }</pre> |
| Description | <p>Provides the information to the client whether an issue is pending on the MR Hosts (isApplicable == true) that currently blocks further actions on the MR Host.</p> <p>To continue execution on / interaction with the MR Host, the client needs to handle the issue via 'handleIssue'.</p> <p>An issue may require the remote client to provide additional parameters regarding how an issue is supposed to be confirmed (also see section 2.7.13.1)</p> <p>The value range of possible parameters that goes along with a handling option id is provided by the "parameterRange" property.</p> |

| | |
|---------------------------|--|
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/issue/getIssue?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"isApplicable":true,"value":{"handlingOptions":[{"id":"d65f21db-e69b-49f4-830b-117cc02996ff"}, {"id":"d65f21db-e69b-49f4-830b-117cc02996ff"}], "additionalInformation":[{"key":"ABC","value":"xyz"}, {"key":"ABC","value":"xyz"}]}, "issueType":"automatictablemove"}</code> |
| WebSocket Message Example | <code>{"service":"product/issue","request":"getIssue","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"isApplicable":true,"value":{"handlingOptions":[{"id":"d65f21db-e69b-49f4-830b-117cc02996ff"}, {"id":"d65f21db-e69b-49f4-830b-117cc02996ff"}], "additionalInformation":[{"key":"ABC","value":"xyz"}, {"key":"ABC","value":"xyz"}]}, "issueType":"automatictablemove"}}</code> |

2.7.13.3.7 handleIssue

| | |
|--------------------|--|
| Service | issue |
| Request | handleIssue |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | <pre>{ "id": "confirm decline", "parameters": [{ "key": "String", "value": "Object" }] }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noIssuePending requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Is called by the client to handle a pending issue. The client must provide an id that reflects one of the handling options that were provided with the issue returned by the getIssue command. For complex issues, the client additionally has to provide valid parameters to specify how to confirm the issue. |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/issue/handleIssue</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"confirm", "parameters":[{"key":"ABC","value":"xyz"}, {"key":"ABC","value":"xyz"}]}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.14 Adjustment Service

2.7.14.1 Introduction

The Adjustment service allows getting and setting of the system frequency, the transmitter reference amplitude, shim currents and gradient offsets. Furthermore, all adjustments may be executed or invalidated via a single command.

Adjustment operations are always related to the currently open protocol / template, i.e. they can only be executed when the corresponding template is open.

2.7.14.2 Adjustment Service Operations

| | |
|---|-----|
| 2.7.14.2.1 setFrequency | 210 |
| 2.7.14.2.2 getFrequency | 210 |
| 2.7.14.2.3 setTransmitterAmplitude | 211 |
| 2.7.14.2.4 getTransmitterAmplitude | 211 |
| 2.7.14.2.5 setAllAdjustments | 212 |
| 2.7.14.2.6 getAllAdjustments | 213 |
| 2.7.14.2.7 adjustAll | 214 |
| 2.7.14.2.8 stopAdjustAll | 214 |
| 2.7.14.2.9 getAdjustAllState | 215 |
| 2.7.14.2.10 invalidateAll | 215 |
| 2.7.14.2.11 connectServiceToDefaultWebSocket | 216 |
| 2.7.14.2.12 connectServiceToWebSocket | 216 |
| 2.7.14.2.13 disconnectServiceFromDefaultWebSocket | 217 |
| 2.7.14.2.14 disconnectServiceFromWebSocket | 218 |
| 2.7.14.2.15 resendServiceStateViaWebSockets | 218 |

2.7.14.2.1 setFrequency

| | |
|--------------------|---|
| Service | adjustment |
| Request | setFrequency |
| Request Properties | Request Type: POST Required Privilege Level : Advanced Requires Host Control |
| Parameters | { "value": "Double" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Set system frequency Prerequisite: Template must be open |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/setFrequency Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": 1.0 } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.14.2.2 getFrequency

| | |
|--------------------|---|
| Service | adjustment |
| Request | getFrequency |
| Request Properties | Request Type: GET Required Privilege Level : Advanced |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" } |
| Description | Get system frequency Prerequisite: Template must be open |

| | |
|---------------|--|
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/adjustment/getFrequency?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":1.0}</code> |

2.7.14.2.3 setTransmitterAmplitude

| | |
|--------------------|---|
| Service | adjustment |
| Request | setTransmitterAmplitude |
| Request Properties | Request Type: POST Required Privilege Level : Advanced Requires Host Control |
| Parameters | <code>{ "value": "Double" }</code> |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</code> |
| Description | Set transmitter amplitude Prerequisite: Template must be open |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/adjustment/setTransmitterAmplitude</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "value":1.0}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.14.2.4 getTransmitterAmplitude

| | |
|--------------------|---|
| Service | adjustment |
| Request | getTransmitterAmplitude |
| Request Properties | Request Type: GET Required Privilege Level : Advanced |
| Parameters | n/a |
| Response | <code>{ "result": { "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning </code> |

| | |
|---------------|---|
| | <pre>requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Double" }</pre> |
| Description | Get transmitter amplitude Prerequisite: Template must be open |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/getTransmitterAmplitude?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":1.0}</pre> |

2.7.14.2.5 setAllAdjustments

| | |
|--------------------|---|
| Service | adjustment |
| Request | setAllAdjustments |
| Request Properties | Request Type: POST Required Privilege Level : Advanced Requires Host Control |
| Parameters | <pre>{ "frequency": "Double", "transmitterAmplitude": "Double", "gradientOffset": { "x": "Double", "y": "Double", "z": "Double" }, "shimCurrents": { "z2": "Double", "zx": "Double", "zy": "Double", "x2y2": "Double", "xy": "Double" } }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Set all adjustments (frequency, transmitter amplitude, shim currents and gradient offsets) Prerequisite: Template must be open |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/setAllAdjustments Request body: <pre>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "frequency":1.0,</pre> |

| | |
|------------------|--|
| | "transmitterAmplitude":1.0, "gradientOffset":{"x":1.0,"y":1.0,"z":1.0}, "shimCurrents":{"z2":1.0,"zx":1.0,"zy":1.0,"x2y2":1.0,"xy":1.0}} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.14.2.6 getAllAdjustments

| | |
|-----------------------|---|
| Service | adjustment |
| Request | getAllAdjustments |
| Request Properties | Request Type: GET Required Privilege Level : Advanced |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "frequency": "Double", "transmitterAmplitude": "Double", "gradientOffset": { "x": "Double", "y": "Double", "z": "Double" }, "shimCurrents": { "z2": "Double", "zx": "Double", "zy": "Double", "x2y2": "Double", "xy": "Double" } }</pre> |
| Description | Get all adjustments (frequency, transmitter amplitude, shim currents and gradient offsets) Prerequisite: Template must be open |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/getAllAdjustments?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"frequency":1.0,"transmitterAmplitude":1.0,"gradientOffset":{"x":1.0,"y":1.0,"z":1.0},"shimCurrents":{"z2":1.0,"zx":1.0,"zy":1.0,"x2y2":1.0,"xy":1.0}} |

2.7.14.2.7 adjustAll

| | |
|--------------------|---|
| Service | adjustment |
| Request | adjustAll |
| Request Properties | Request Type: POST Required Privilege Level : Advanced Requires Host Control |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Adjust all Prerequisite: Template must be open |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/adjustAll Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.14.2.8 stopAdjustAll

| | |
|--------------------|---|
| Service | adjustment |
| Request | stopAdjustAll |
| Request Properties | Request Type: POST Required Privilege Level : Advanced |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noAdjustmentOperationInProgress timeout requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Stops the currently running adjustments that were started via AdjustAll Prerequisite: Template must be open |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/stopAdjustAll Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |

| | |
|---------------|---|
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |
|---------------|---|

2.7.14.2.9 getAdjustAllState

| | |
|---------------------------|--|
| Service | adjustment |
| Request | getAdjustAllState |
| Request Properties | Request Type: GET Required Privilege Level : Advanced Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "undefined running issuePending finishedWithSuccess finishedWithError" }</pre> |
| Description | Represents the state of the AdjustAll operation, which can be: <ul style="list-style-type: none"> - undefined - running - issuePending - finishedWithSuccess - finishedWithFailure Prerequisite: Template must be open |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/getAdjustAllState?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"undefined"}</pre> |
| WebSocket Message Example | <pre>{"service":"product/adjustment","request":"getAdjustAllState","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"undefined"}}</pre> |

2.7.14.2.10 invalidateAll

| | |
|--------------------|--|
| Service | adjustment |
| Request | invalidateAll |
| Request Properties | Request Type: POST Required Privilege Level : Advanced Requires Host Control |
| Parameters | n/a |
| Response | <pre>{ "result": {</pre> |

| | |
|---------------|---|
| | <pre> "success": "Boolean", "reason": "ok unknownError issuePending adjustmentOperationInProgress noTemplateOpen templateRunning requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } </pre> |
| Description | <p>Invalidate all</p> <p>Prerequisite: Template must be open</p> |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/adjustment/invalidateAll</p> <p>Request body:</p> <pre> {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763"} </pre> |
| Resp. Example | <pre> {"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}} </pre> |

2.7.14.2.11 connectServiceToDefaultWebSocket

| | |
|--------------------|---|
| Service | adjustment |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | <p>Request Type: POST</p> <p>Required Privilege Level : Read</p> |
| Parameters | n/a |
| Response | <pre> { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } </pre> |
| Description | <p>Enables Websockets for this service</p> <p>Prerequisites:</p> <ul style="list-style-type: none"> - none |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/adjustment/connectServiceToDefaultWebSocket</p> <p>Request body:</p> <pre> {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763"} </pre> |
| Resp. Example | <pre> {"result": {"success": true, "reason": "ok", "time": "20170608T143325.423"}} </pre> |

2.7.14.2.12 connectServiceToWebSocket

| | |
|---------|---------------------------|
| Service | adjustment |
| Request | connectServiceToWebSocket |

| | |
|--------------------|--|
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/connectServiceToWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.14.2.13 disconnectServiceFromDefaultWebSocket

| | |
|--------------------|--|
| Service | adjustment |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/disconnectServiceFromDefaultWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

| | |
|---------|--|
| Example | |
|---------|--|

2.7.14.2.14 disconnectServiceFromWebSocket

| | |
|--------------------|--|
| Service | adjustment |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/adjustment/disconnectServiceFromWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.14.2.15 resendServiceStateViaWebSockets

| | |
|--------------------|--|
| Service | adjustment |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |

| | |
|---------------|--|
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | <code>https://125.28.36.11:7787/SRC/v2/product/adjustment/resendServiceStateViaWebSockets</code> Request body: <code>{"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</code> |
| Resp. Example | <code>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</code> |

2.7.15 Parameter/configured Service

2.7.15.1 Introduction

The Configured Parameter Service (`parameter/configured`) allows the remote access to the protocol parameters of a protocol step that have been configured in the configurable parameter card within the AddIn of a template step.

Currently, configured parameters of the following types are supported:

- BOOL
- LONG
- DOUBLE
- CHOICE

The RC has access to

- all static parameter metadata:
 - Unique ID
 - Label
 - Type
 - Unit
 - Description
- dynamic parameter metadata
 - Parameter limits
 - In case of a choice parameter (a parameter that accepts 1 of n possible concrete – mostly textual - values) the set of possible parameter values
 - Parameter availability
 - Parameter editability
- parameter value.

2.7.15.2 Reading and writing a parameter's value

The parameters and their metadata may be retrieved and changed only of the currently open protocol. In general, every step may have a different set of preconfigured parameters (due to different configurations in the configurable parameter card of the step).

2.7.15.3 How to use the Configured Parameter Service

The client of the Configured Parameter Service should adhere to the following approach:

1. Retrieving the set of available parameters (i.e. parameters that have been configured within the configurable parameter card within the step's AddIn configuration) when a template opens:

`getConfiguredParameters`

2. As protocol parameters and some of their properties may change over time on the scanner while the protocol is open, the client should poll the following data in a regular interval (e.g. every 1-2 seconds):

- Parameter Value: `getParameterValue`
- Parameter Limits (only if really needed): `getParameterLimits`
- Parameter's choice values (if applicable): `getParameterChoices`
- The parameter's current state: `getIsParameterAvailable`
`getIsParameterEditable`

All the operations mentioned above – except for `getParameterLimits` - are also supported by WebSockets, so the best approach also here is to wait for status changes of `getConfiguredParameters`, `getParameterValue`, `getParameterChoices`, `getIsParameterAvailable`, `getIsParameterEditable`.

As retrieving the limits of a particular parameter may have a significant performance penalty this is not supported automatically by WebSockets.

3. The parameter may be changed using `setParameterValue`

In other words, the RC implementation can be completely data-driven by WebSocket messages except for retrieving a parameter's limit and changing its value.

2.7.15.4 Parameter/configured Service Operations

| | |
|---|-----|
| 2.7.15.4.1 getConfiguredParameters..... | 223 |
| 2.7.15.4.2 getParameterInfo | 224 |
| 2.7.15.4.3 getParameterValue..... | 224 |
| 2.7.15.4.4 getParameterDescription | 225 |
| 2.7.15.4.5 getParameterChoices | 226 |
| 2.7.15.4.6 setParameterValue | 226 |
| 2.7.15.4.7 getIsParameterAvailable | 228 |
| 2.7.15.4.8 getIsParameterEditable | 228 |
| 2.7.15.4.9 getParameterLimits..... | 229 |
| 2.7.15.4.10 connectServiceToDefaultWebSocket..... | 230 |
| 2.7.15.4.11 connectServiceToWebSocket..... | 230 |
| 2.7.15.4.12 disconnectServiceFromDefaultWebSocket | 231 |
| 2.7.15.4.13 disconnectServiceFromWebSocket | 231 |
| 2.7.15.4.14 resendServiceStateViaWebSockets..... | 232 |

2.7.15.4.1 getConfiguredParameters

| | |
|---------------------------|--|
| Service | parameter/configured |
| Request | getConfiguredParameters |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen noProtocolAvailable parameterNotAvailable noChoiceParameter unsupportedParameterType requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": [{ "index": "Int32", "id": "Guid", "label": "String", "type": "String", "unit": "String", "ProtocolTag": "String" }] }</pre> |
| Description | <p>Each parameter that is returned from getConfiguredParameters is represented by a parameter info structure that contains the following information:</p> <ul style="list-style-type: none"> - id: The parameter's unique id the client has to use for addressing a parameter - label: The parameter's label - type: The parameter's type which may be BOOL, LONG, DOUBLE, CHOICE - unit: The parameter value's unit - description: Detailed description of the parameter <p>As the index of each parameter info within the returned list corresponds to the position of the parameter on the configurable parameter card, the parameter info may also be represented by null if no parameter is configured at a specific position.</p> |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/getConfiguredParameters?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":[{"index":1,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"ABC","type":"ABC","unit":"ABC","ProtocolTag":"ABC"}, {"index":1,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"ABC","type":"ABC","unit":"ABC","ProtocolTag":"ABC"}]}</pre> |
| WebSocket Message Example | <pre>{"service":"product/parameter/configured","request":"getConfiguredParameters","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":[{"index":1,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"ABC","type":"ABC","unit":"ABC","ProtocolTag":"ABC"}, {"index":1,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"ABC","type":"ABC","unit":"ABC","ProtocolTag":"ABC"}]}}</pre> |

2.7.15.4.2 getParameterInfo

| | |
|--------------------|--|
| Service | parameter/configured |
| Request | getParameterInfo |
| Request Properties | Request Type: GET Required Privilege Level : Execute Requires Host Control |
| Parameters | { "protocolTag": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen noProtocolAvailable parameterNotAvailable noChoiceParameter unsupportedParameterType requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "id": { "index": "Int32", "id": "Guid", "label": "String", "type": "String", "unit": "String", "ProtocolTag": "String" } } |
| Description | Returns the parameter info of an arbitrary protocol parameter that is specified by the provided protocol tag. If the protocol tag is unknown or the protocol does not provide the requested parameter, null is returned. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/getParameterInfo?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&protocolTag=ABC |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"id":{"index":1,"id":"26eb060c-553d-4c15-bf5b-23cf29012763","label":"ABC","type":"ABC","unit":"ABC","ProtocolTag":"ABC"}} |

2.7.15.4.3 getParameterValue

| | |
|--------------------|--|
| Service | parameter/configured |
| Request | getParameterValue |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | { "id": "Guid" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen noProtocolAvailable parameterNotAvailable noChoiceParameter unsupportedParameterType " |

| | |
|---------------------------|---|
| | <pre>requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Object" }</pre> |
| Description | Returns the value of a protocol parameter specified by id (the unique id of a parameter that is contained in the parameter info that has been retrieved by getConfiguredParameters). Semantically, the value can be a BOOL, LONG, DOUBLE or CHOICE, depending on the parameter's type. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/getParameterValue?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"xyz"}</pre> |
| WebSocket Message Example | <pre>{"service":"product/parameter/configured","request":"getParameterValue","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"xyz"}}</pre> |

2.7.15.4.4 getParameterDescription

| | |
|---------------------------|--|
| Service | parameter/configured |
| Request | getParameterDescription |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | <pre>{ "id": "Guid" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen noProtocolAvailable parameterNotAvailable noChoiceParameter unsupportedParameterType requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "String" }</pre> |
| Description | Returns the description of the protocol parameter |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/getParameterDescription?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"ABC"}</pre> |
| WebSocket Message Example | <pre>{"service":"product/parameter/configured","request":"getParameterDescription","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"ABC"}}</pre> |

2.7.15.4.5 getParameterChoices

| | |
|---------------------------|--|
| Service | parameter/configured |
| Request | getParameterChoices |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | { "id": "Guid" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen noProtocolAvailable parameterNotAvailable noChoiceParameter unsupportedParameterType requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": ["String"] } |
| Description | Retrieves all possible values of a CHOICE parameter. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/getParameterChoices?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":["ABC","ABC"]} |
| WebSocket Message Example | {"service":"product/parameter/configured","request":"getParameterChoices","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":["ABC","ABC"]}} |

2.7.15.4.6 setParameterValue

| | |
|--------------------|--|
| Service | parameter/configured |
| Request | setParameterValue |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | { "id": "Guid", "value": "Object", "allowSideEffects": "Boolean" } |
| Response | { "result": { "success": "Boolean", "reason": "ok protocolNotAvailable parameterNotAvailable parameterNotEditable wrongValueFormat parameterChangeWouldCauseSideEffects unknownError requiresHostControl" } } |

| | |
|---------------|--|
| | <pre> noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "valueSet": "Object" }</pre> |
| Description | <p>Sets the value of a parameter that is specified by id.</p> <p>Semantically, the value can be either BOOL, LONG, DOUBLE or CHOICE, depending on the parameter the value is set to.</p> <p>In general, in an MR protocol many parameters have dependencies among each other, meaning that changing one parameter may result in a change of one or multiple other parameters.</p> <p>Therefore the SetParameterValue operation has an additional boolean parameter 'allowSideEffects'. If this is set to true, the parameter will be set to the desired value irrespective of changes of other parameters (meaning that the remote client automatically accepts all induced changes of other parameters).</p> <p>If 'allowSideEffects' is set to false and a change in the parameter would induce other parameter changes, the set operation fails, the protocol parameter remains unchanged and the result reason 'ParameterChangeWouldCauseSideEffects' will be returned to the remote client.</p> <p>Furthermore - and independent of the 'allowSideEffects' parameter - there are protocol parameters that only allow certain concrete values or/and have a maximum and a minimum value.</p> <p>If the value to be set cannot be set exactly due to these constraints it will be set to the next possible value (meaning the next possible concrete value or - in case the value set is above the allowed maximum - to the maximum value) and the set operation succeeds.</p> <p>To allow the remote client to check this incident each set operation has a 'valueSet' return value that shows which value actually was set, so the remote client can see whether the supposed and actual values differ.</p> |
| Req. Example | <p>https://125.28.36.11:7787/SRC/v2/product/parameter/configured/setParameterValue</p> <p>Request body:</p> <pre>{ "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "26eb060c-553d-4c15-bf5b-23cf29012763", "value": "xyz", "allowSideEffects": true }</pre> |
| Resp. Example | <pre>{ "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" }, "valueSet": "xyz" }</pre> |

2.7.15.4.7 getIsParameterAvailable

| | |
|---------------------------|---|
| Service | parameter/configured |
| Request | getIsParameterAvailable |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | { "id": "Guid" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen noProtocolAvailable parameterNotAvailable noChoiceParameter unsupportedParameterType requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "Boolean" } |
| Description | Tells whether the specified parameter is available. In certain circumstances a parameter may become temporarily or completely unavailable. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/getIsParameterAvailable?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":true} |
| WebSocket Message Example | {"service":"product/parameter/configured","request":"getIsParameterAvailable","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":true}} |

2.7.15.4.8 getIsParameterEditable

| | |
|--------------------|---|
| Service | parameter/configured |
| Request | getIsParameterEditable |
| Request Properties | Request Type: GET Required Privilege Level : Execute Supported by WebSockets |
| Parameters | { "id": "Guid" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen noProtocolAvailable parameterNotAvailable noChoiceParameter unsupportedParameterType requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } |

| | |
|---------------------------|---|
| | <pre>}, "value": "Boolean" }</pre> |
| Description | Tells whether the specified parameter is editable. In certain circumstances a parameter may become temporarily or completely read-only. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/getIsParameterEditable?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":true}</pre> |
| WebSocket Message Example | <pre>{"service":"product/parameter/configured","request":"getIsParameterEditable","response":{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":true}}</pre> |

2.7.15.4.9 getParameterLimits

| | |
|--------------------|--|
| Service | parameter/configured |
| Request | getParameterLimits |
| Request Properties | Request Type: GET Required Privilege Level : Execute |
| Parameters | <pre>{ "id": "Guid" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError noTemplateOpen noProtocolAvailable parameterNotAvailable noChoiceParameter unsupportedParameterType requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": [{ "min": "Double", "max": "Double", "increment": "Double", "restriction": "String", "interval": "String" }] }</pre> |
| Description | Returns the list of limits for the specified parameter. |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/getParameterLimits?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763&id=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":[{"min":1.0,"max":1.0,"increment":1.0,"restriction":"ABC","interval":"ABC"}, {"min":1.0,"max":1.0,"increment":1.0,"restriction":"ABC","interval":"ABC"}]}</pre> |

2.7.15.4.10 connectServiceToDefaultWebSocket

| | |
|--------------------|--|
| Service | parameter/configured |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/connectServiceToDefaultWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.15.4.11 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | parameter/configured |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/connectServiceToWebSocket |

| | |
|------------------|---|
| | Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.15.4.12 disconnectServiceFromDefaultWebSocket

| | |
|-----------------------|--|
| Service | parameter/configured |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/parameter/configured/disconnectServiceFromDefaultWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.15.4.13 disconnectServiceFromWebSocket

| | |
|-----------------------|---|
| Service | parameter/configured |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered |

| | |
|---------------|--|
| | <pre>clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/parameter/configured/disconnectServiceFromWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.15.4.14 resendServiceStateViaWebSockets

| | |
|--------------------|---|
| Service | parameter/configured |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/parameter/configured/resendServiceStateViaWebSockets Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.16 Debugging Service

2.7.16.1 Introduction

This service allows the RC to trigger the creation and retrieval of the MR savelog generation. The savelog generation can require several minutes to succeed.

2.7.16.2 Debugging Service Operations

| | |
|--|-----|
| 2.7.16.2.1 createAndRetrieveSavelog | 234 |
| 2.7.16.2.2 connectServiceToDefaultWebSocket..... | 235 |
| 2.7.16.2.3 connectServiceToWebSocket..... | 235 |
| 2.7.16.2.4 disconnectServiceFromDefaultWebSocket | 236 |
| 2.7.16.2.5 disconnectServiceFromWebSocket | 236 |
| 2.7.16.2.6 resendServiceStateViaWebSockets | 237 |

2.7.16.2.1 createAndRetrieveSavelog

| | |
|--------------------|--|
| Service | debugging |
| Request | createAndRetrieveSavelog |
| Request Properties | Request Type: POST Required Privilege Level : Execute Requires Host Control |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | <p>Trigger creation of debug information on the host (trigger savelog generation).</p> <p>The HTTP call will return when the savelog creation was initiated on the MR Host.</p> <p>While the savelog is being created the MR Host is in a "Maintenance Mode", i.e. during this time all HTTP requests (except for remote/getIsActive and authorization/getIsRegistered) will fail giving the reason "maintenanceActive". This is because no further actions on the MR Host should interfere with the savelog generation.</p> <p>The actual savelog (zip-file) will then be sent as a series of binary websocket messages, surrounded by textual meta websocket messages.</p> <p>The implementation on the client side should proceed as follows upon receiving WebSocket messages:</p> <p>MESSAGE: {"service":"product/debugging","request":"savelogStreamStart","response":{}}</p> <p>=> Create new zip-file on client side into which the upcoming binary messages are written to</p> <p>... series of binary messages that must be sequentially written into the zip file ...</p> <p>MESSAGE: {"service":"product/debugging","request":"savelogStreamEnd","response":{}}</p> <p>=> Close zip-file on client side, the file should be complete at that point.</p> <p>The savelog generation will only be possible if a patient is registered and the calling remote client has host control.</p> |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/debugging/createAndRetrieveSavelog Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Exam. | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.16.2.2 connectServiceToDefaultWebSocket

| | |
|--------------------|---|
| Service | debugging |
| Request | connectServiceToDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/debugging/connectServiceToDefaultWebSocket Request body: {"sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763"} |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}} |

2.7.16.2.3 connectServiceToWebSocket

| | |
|--------------------|---|
| Service | debugging |
| Request | connectServiceToWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | <pre>{ "id": "String" }</pre> |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Enables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/debugging/connectServiceToWebSocket |

| | |
|------------------|---|
| | Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763", "id": "d65f21db-e69b-49f4-830b-117cc02996ff" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.16.2.4 disconnectServiceFromDefaultWebSocket

| | |
|-----------------------|--|
| Service | debugging |
| Request | disconnectServiceFromDefaultWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } } |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/debugging/disconnectServiceFromDefaultWebSocket Request body: { "sessionId": "26eb060c-553d-4c15-bf5b-23cf29012763" } |
| Resp. Example | { "result": { "success": true, "reason": "ok", "time": "20170608T143325.423" } } |

2.7.16.2.5 disconnectServiceFromWebSocket

| | |
|-----------------------|---|
| Service | debugging |
| Request | disconnectServiceFromWebSocket |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | { "id": "String" } |
| Response | { "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered |

| | |
|---------------|---|
| | <pre>clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Disables Websockets for this service Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/debugging/disconnectServiceFromWebSocket Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763", "id":"d65f21db-e69b-49f4-830b-117cc02996ff"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.16.2.6 resendServiceStateViaWebSockets

| | |
|--------------------|---|
| Service | debugging |
| Request | resendServiceStateViaWebSockets |
| Request Properties | Request Type: POST Required Privilege Level : Read |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" } }</pre> |
| Description | Resends the service's state via WebSocket to the requesting client Prerequisites: - none |
| Req. Example | <pre>https://125.28.36.11:7787/SRC/v2/product/debugging/resendServiceStateViaWebSockets Request body: {"sessionId":"26eb060c-553d-4c15-bf5b-23cf29012763"}</pre> |
| Resp. Example | <pre>{"result":{"success":true,"reason":"ok","time":"20170608T143325.423"}}</pre> |

2.7.16.2.7 Access-i Logging

All relevant communication between RCs and the MR Host with respect to controlling the MR workflow is logged on the MR host. It cannot be retrieved from the RC remotely, but is contained within the MR savelog package. The logging information is comprised by the MR examination application log file to provide all relevant actions of the ongoing MR workflow.

2.7.17 Time Service

2.7.17.1 Introduction

Provides the date and time on the MR Host to the RC.

2.7.17.2 Time Service Operations

2.7.17.2.1 getCurrentDateTime238

2.7.17.2.1 getCurrentDateTime

| | |
|--------------------|--|
| Service | time |
| Request | getCurrentDateTime |
| Request Properties | Request Type: GET Required Privilege Level : Basic |
| Parameters | n/a |
| Response | <pre>{ "result": { "success": "Boolean", "reason": "ok unknownError requiresHostControl noClientConnectionsAllowed clientNotRegistered clientPrivilegeInsufficient maintenanceActive", "time": "DateTime" }, "value": "String" }</pre> |
| Description | |
| Req. Example | https://125.28.36.11:7787/SRC/v2/product/time/getCurrentDateTime?sessionId=26eb060c-553d-4c15-bf5b-23cf29012763 |
| Resp. Example | {"result":{"success":true,"reason":"ok","time":"20170608T143325.423"},"value":"ABC"} |

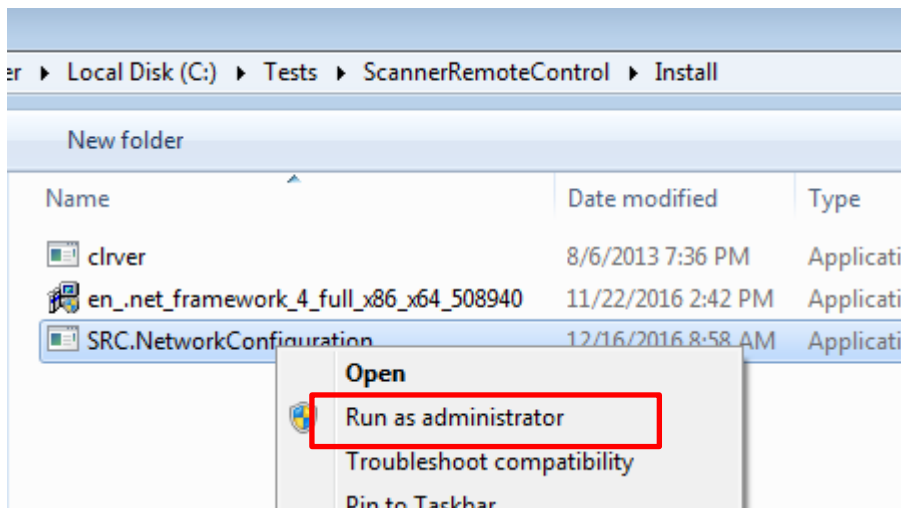
3 Software Development Kit

A test environment is provided for more elaborate tests and to ease the integration and testing of remote clients with Access-i. It comprises an Access-i Simulator and Test Client, which are designed for a Windows PC environment.

3.1 Installation

1. Make sure your operating system is Windows 7 or higher
2. Ensure that .NET Framework 4.7 or higher installed.
3. Copy the root folder of the package to a local hard drive on your machine.
4. Run AccessI.NetworkConfiguration.exe as Administrator:

- Right-Click on file \Install\AccessI.NetworkConfiguration.exe and click "Run as administrator":



Depending on the privileges of your Window account you may be asked the name and password of an Administrator's account.

3.2 Access-i Simulator

The Access-i Simulator shall enable the RC implementation and testing without having physical access to a MR host and scanner. It aims to mimic the functionality of the MR host which is associated with Access-i, i.e. the loading, execution and modification of template steps and the retrieval of MR images and projection data. In addition, all communication between the Access-i Simulator and RC is logged on the screen (Figure 24) and in a log file in the “log” subfolder.

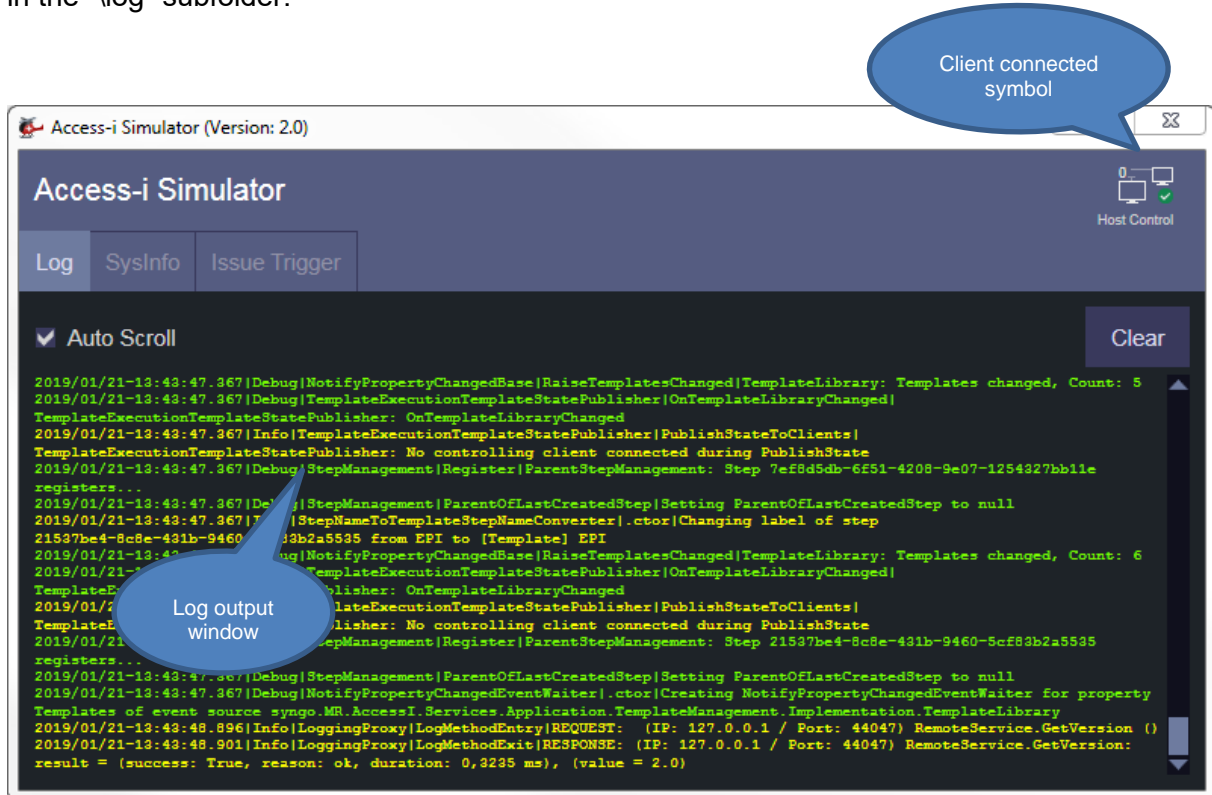


Figure 24: Access-i Simulator log window for offline development and testing of RCs.

The Access-i Simulator consists of 4 parts:

- A headline with the connected icons that tells whether a client is connected.
- A log output window that displays what happens on the Access-i simulator.
- A dedicated tab to simulate different isocenter positions
- A dedicated tab to simulate different issues.

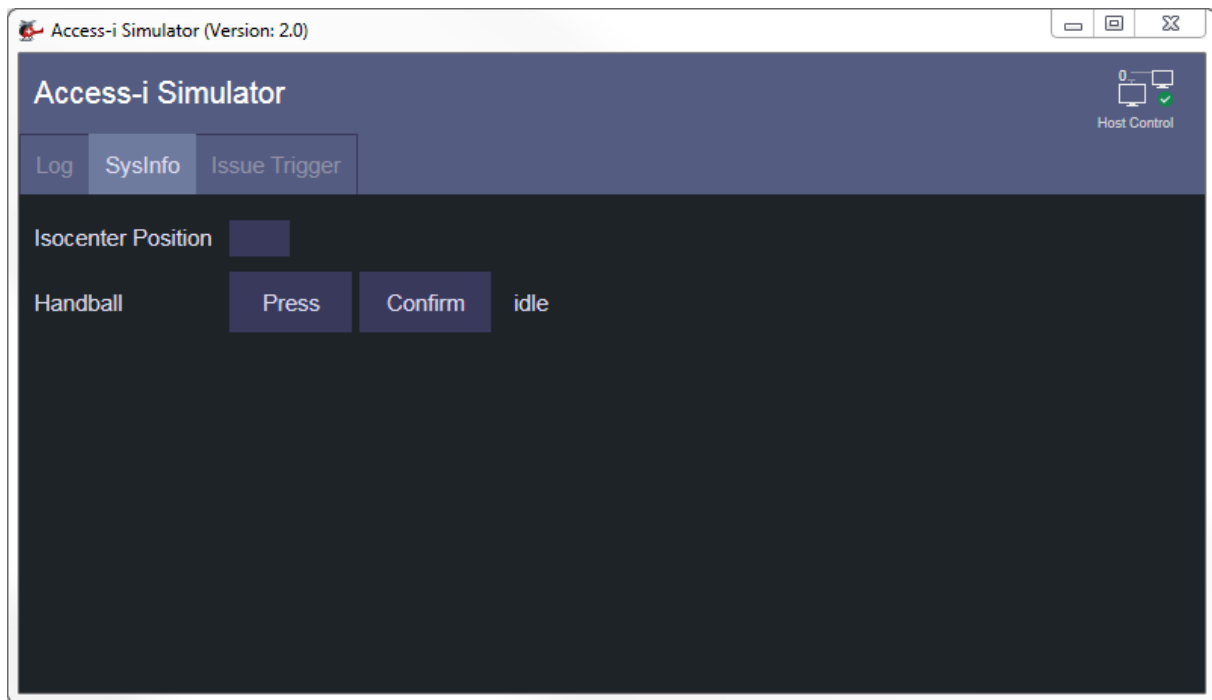


Figure 25: Access-i Simulator system info tab to manipulate the isocenter position.



Figure 26: Access-i Simulator Issue Trigger.

3.3 Access-i Test Client

The Access-i test client shall provide an exemplary implementation of a RC, but also serve as an official test tool to verify the provided functionality of the Access-i interface. It provides a graphical user interface to load, execute and modify template steps, to receive images and projection data and to retrieve patient data. It is meant to be a supportive tool that acts as “golden reference” of what happens on the Access-i Simulator. The developer of an Access-i client application should always run the test client in parallel to his own client application to compare the behavior of his client and the test client.

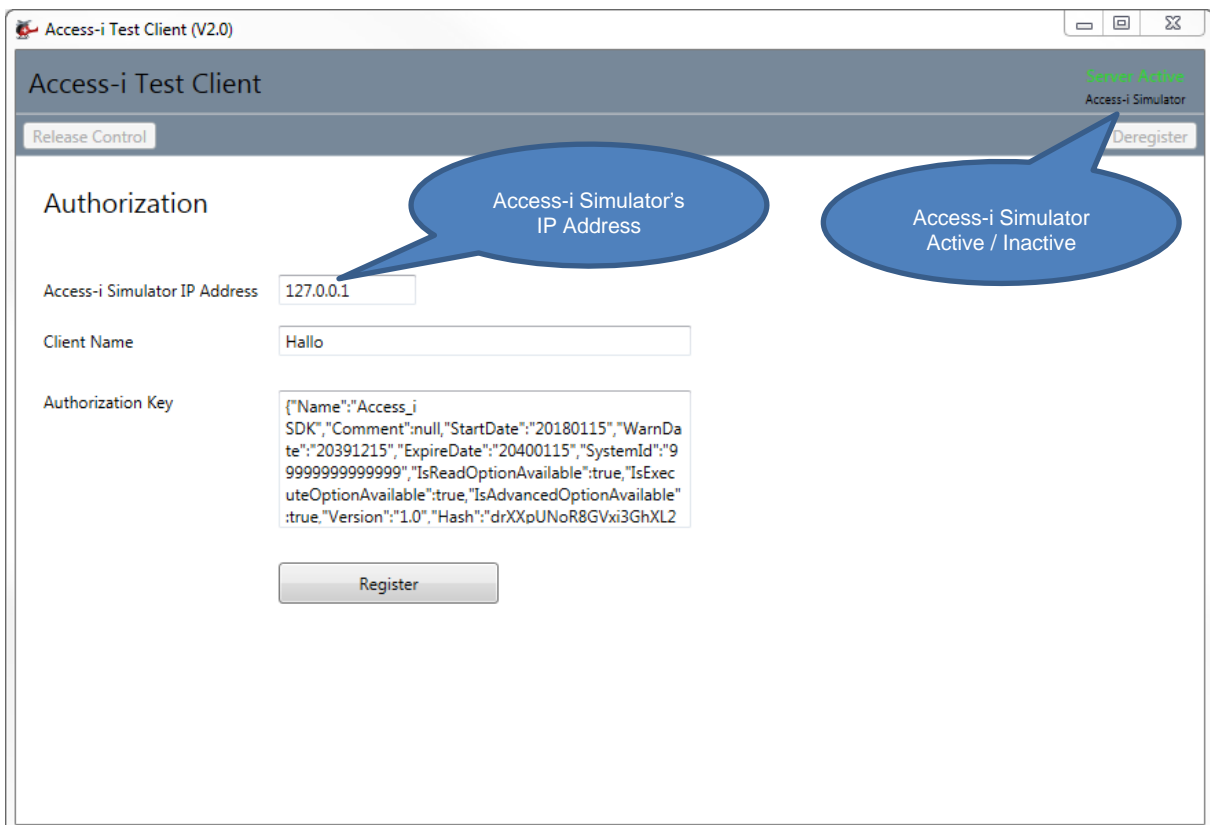


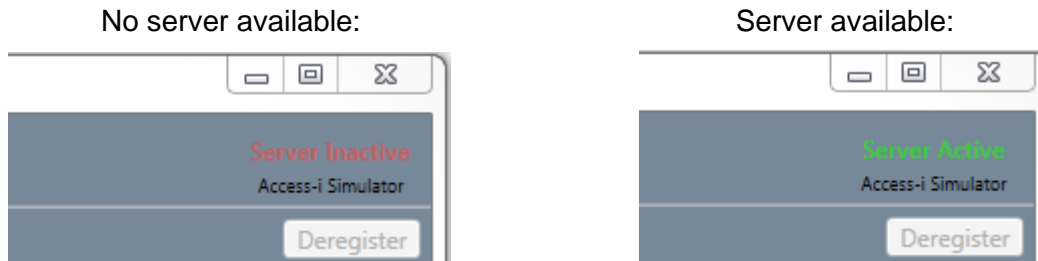
Figure 27: Access-i Test Client with initial UI for authorization and Access-i Simulator IP settings.

The IP address of the Access-i Server (= Access-i Simulator) (here this is localhost 127.0.0.1 which is the default and can always be used if client and server are running of the same machine). Use the “Access-i Simulator IP Address field” to change the target IP address.

Most services offered by Access-i can be triggered via the Access-i Test Client. Furthermore, the state of the Access-i services is shown in several controls which are shortly described in this chapter. To get more details about services please have a look at the corresponding service description chapters.

3.3.1 Remote Service

Via the Remote Service the client knows whether the Access-i Server is active. This is shown via the “Server Inactive” / “Server Active” text top right in the UI:



3.3.2 Authorization Service

Before the client can communicate with the server it needs to register via the Authorization Service and a valid authentication key. Depending on the success of the registration, the UI of the Access-i Test Client will change and offer all available Access-i services from the Access-i Simulator.

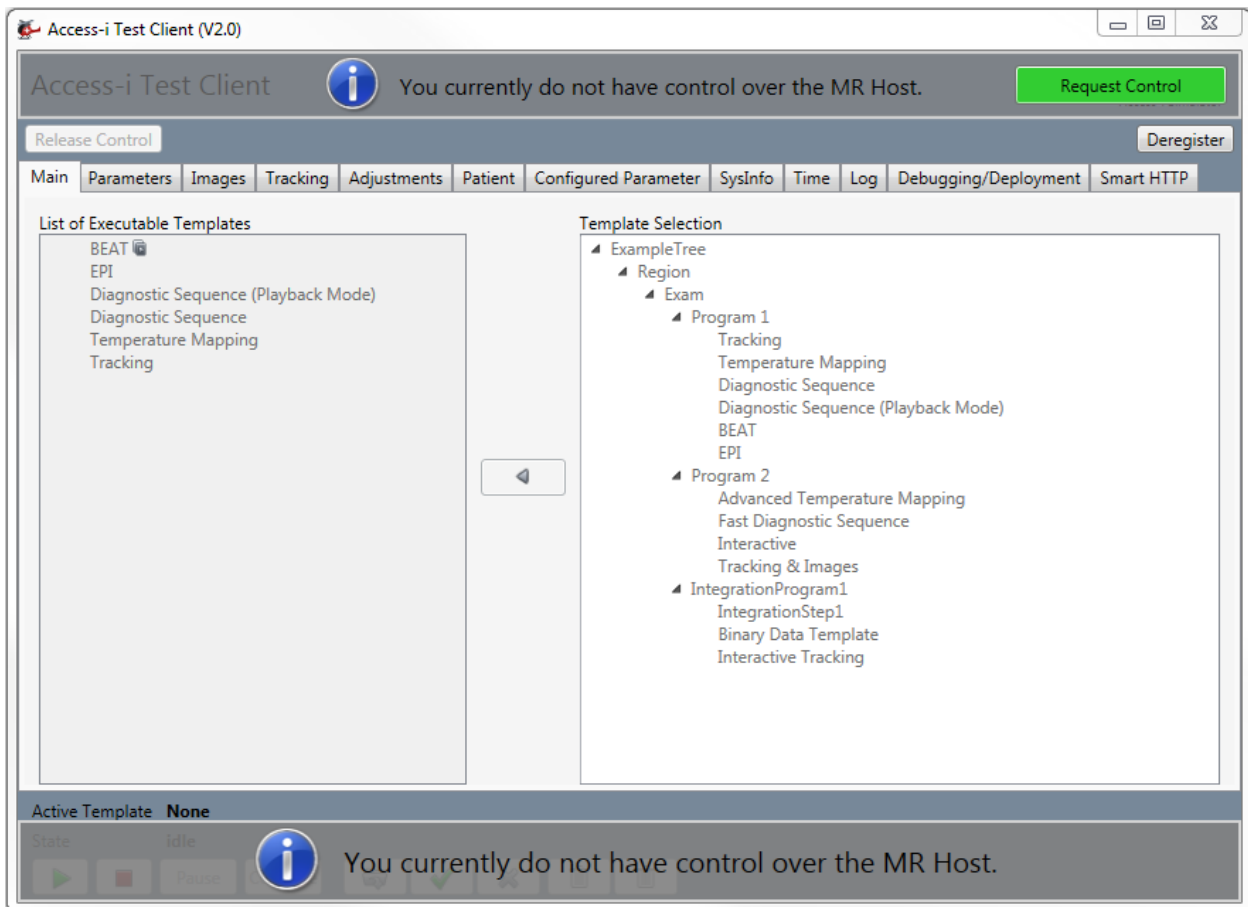


Figure 28: Access-i Test Client after successful authorization and registration.

3.3.3 Host Control Service

After the client has successfully registered, it may retrieve host control and release it. If active host control is not possible, both buttons 'release control' and 'request control' are greyed out.

| | |
|-----------------------------------|---|
| RC can retrieve active control | MR Host. Request Control |
| RC has active control | Access-i Test Client Client is in control Release Control |
| RC cannot retrieve active control | Control cannot be requested because another client is in control. Request Control |

Table 2: Overview of host control states in test client.

The reason for active control is to trigger any active operations via Access-i (like e.g. opening template, starting template etc.).

After your client has requested control, no other client can request control until you release it again.

3.3.4 Template Selection

Via the Template Selection Service you can "Drag & Drop" programs or single templates available in a program tree on the host into the Exam Queue at the scanner. The available programs are shown on the right side of the Main tab:

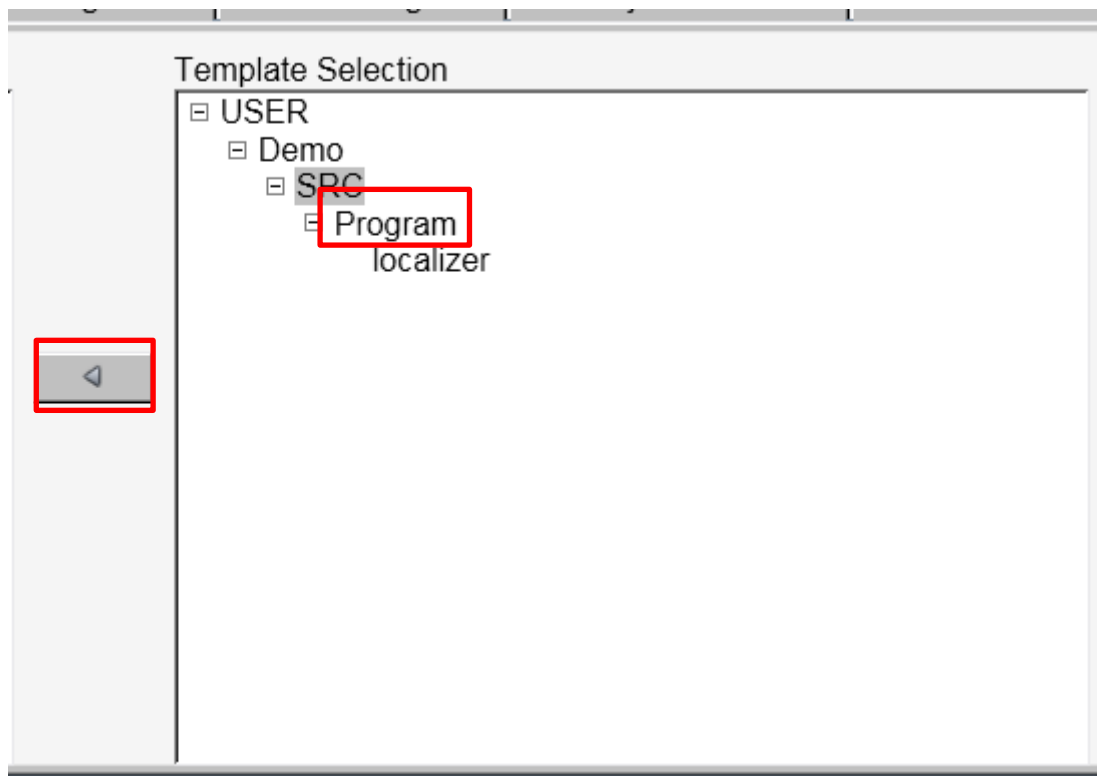


Figure 29: Test Client Template Selection.

To “Drag & Drop” a program into the queue, select the program first and then click the “Left arrow” button.

At startup time of the Access-i Simulator the first program in the program tree is automatically loaded into the queue, this is why multiple templates are already visible in the queue after you have successfully registered:

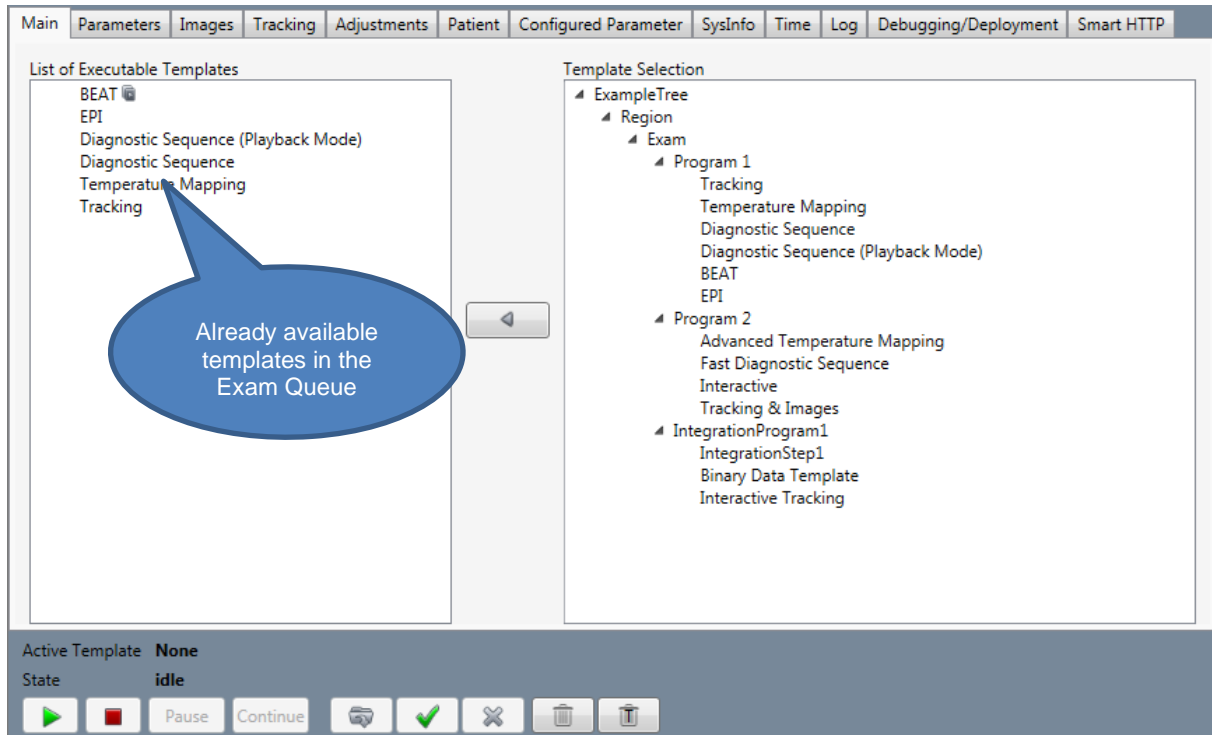


Figure 30: Template selection with test client.

3.3.5 Template Modification

Via the Template Modification Service you can open and close templates. To open a

template you have to select it in the ‘List of Executable Templates’ and then click  :

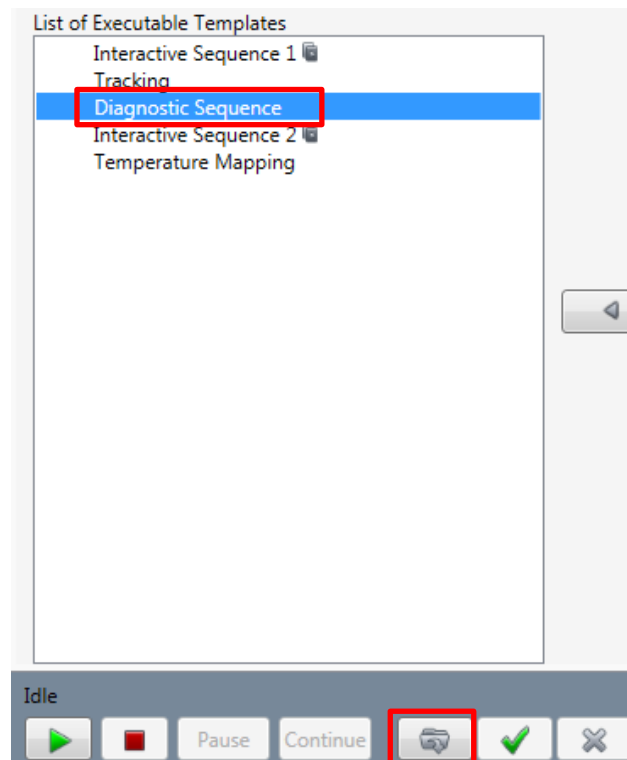




Figure 31: Opening a protocol with test client.

If you want to close an open template, you either click  to accept changes that were made in the protocol while it was open or click  to discard changed:

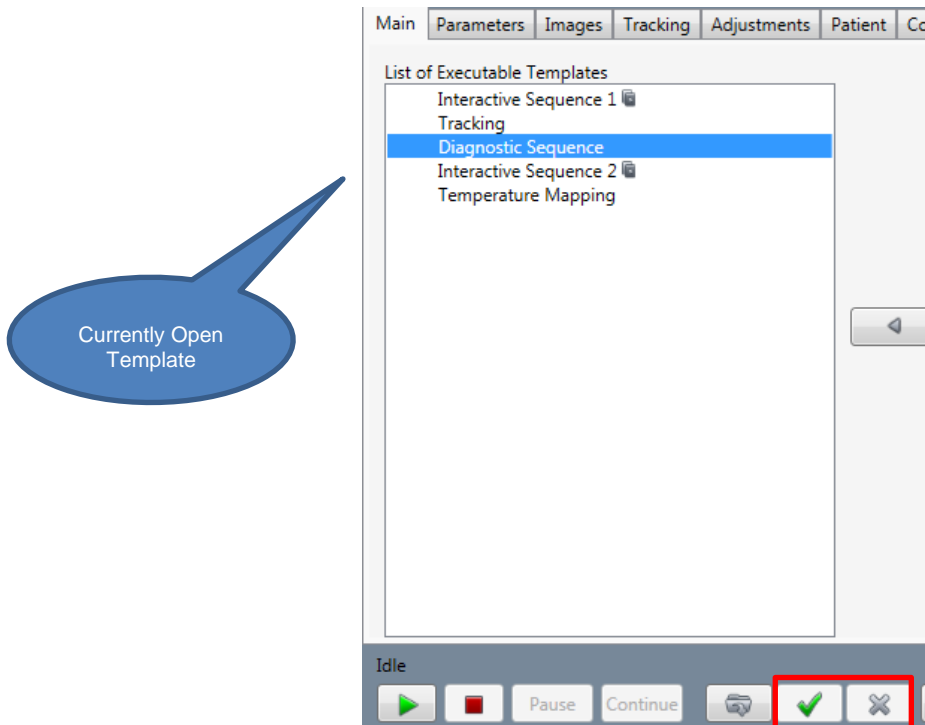


Figure 32: Apply or cancel protocol changes to opened protocol.

3.3.6 Template Execution




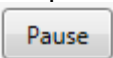
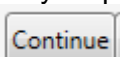
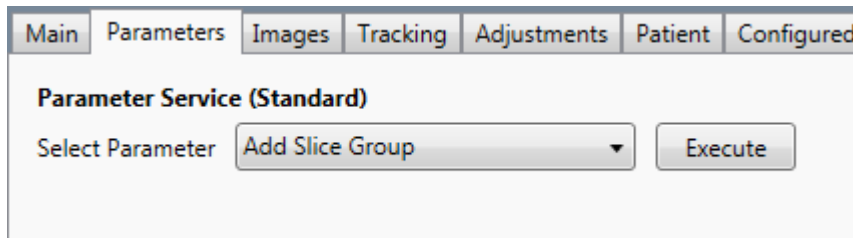
Via the Template Execution Service you Start / Stop / Pause and Continue a Template. First choose the Template to start and press . While the Template is running () , you may stop () or pause () it. When the Template is paused, you may continue () it:



Figure 33: Template execution with test client.

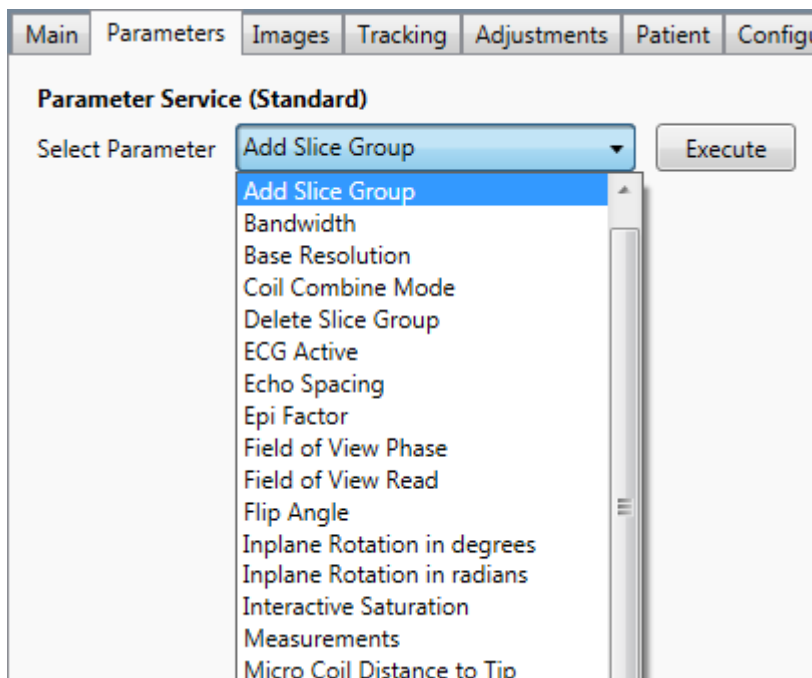
3.3.7 Standard Parameter Service

The Standard Parameter Service is represented in the “Parameters” Tab:

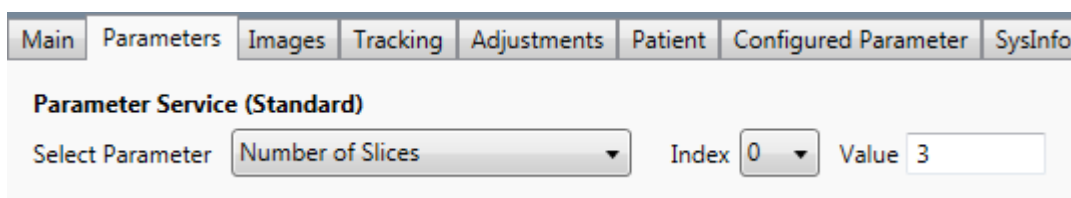


A template must be opened before the Standard Parameter Service can be used.

The UI only allows the adaption of a subset of parameters that can be retrieved and changed via this service:



To show / change e.g. the number of slices, select the entry “Number of Slices” and type in a new value:



3.3.8 Table Service

The Table Service UI is shown in the “Table” Tab:

| Main | Parameters | Images | Tracking | Adjustments | Patient | Config. Parameter | Table | SysInfo |
|---|------------|--------|----------|-------------|---------|-------------------|-------|---------|
| <p>Current Table Position 181</p> <p>Table Positioning Mode <input type="text" value="localRange"/></p> <p>Local Range Position <input type="text" value="0"/></p> <p>Fix Table Position <input type="text" value="0"/></p> | | | | | | | | |

The UI shows the current table position which is immediately updated when the table moves.

You can choose between three global Table Positioning Modes (“*isoCenter*”, “*localRange*”, “*fix*”) and define the **Local Range Position** value which is effective if “*localRange*” table positioning mode is active and the **Fix Table Position** value which is used if “*fix*” mode is selected.

3.3.9 Image Service

The Image Service is represented in the “Images” Tab. This tab shows images after a template that produces images has run. The Access-i Test Client automatically switches to the “Images” tab as soon as images are available. On the image tab the user can choose between the two available image formats, i.e. Dicom and raw16bit images. To differentiate between the two formats, the latter is artificially colored in red.

Note: Be aware that you have to select the desired image type before the scan is started.



Figure 34: Example of a received Dicom Image in the Access-i Test Client

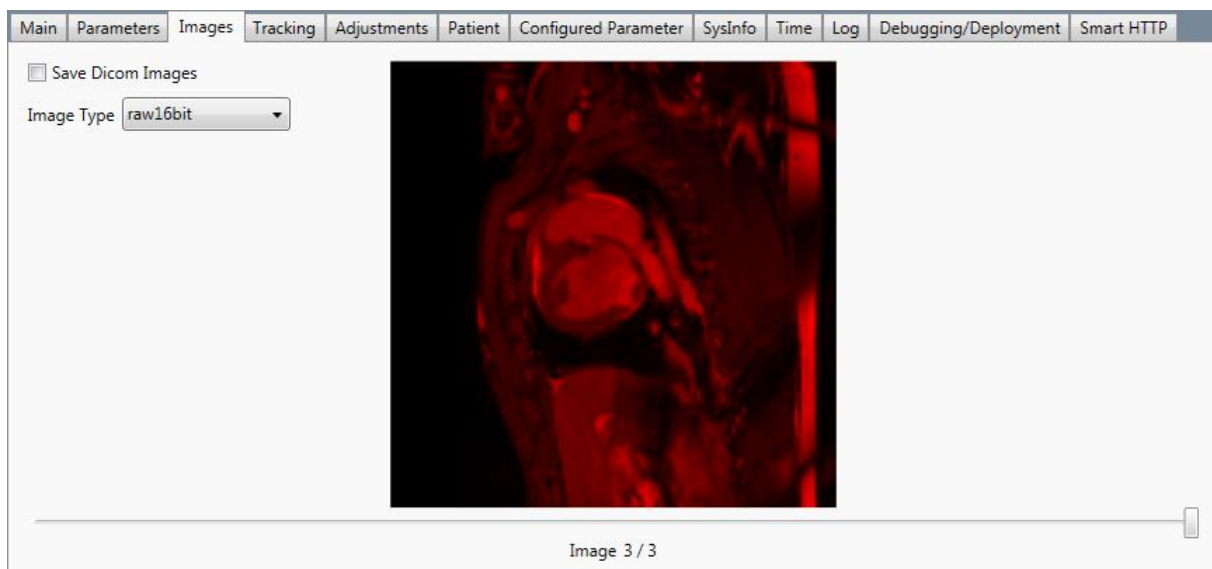


Figure 35: Example of received Raw-16bit image in the Access-i Test Client (shown in red)

3.3.10 Tracking Service

Tracking data is represented in the “Tracking” Tab. It is shown after a tracking template has run. All projections per coil are shown in the tab. The user can switch between coils using the navigation buttons at the top.

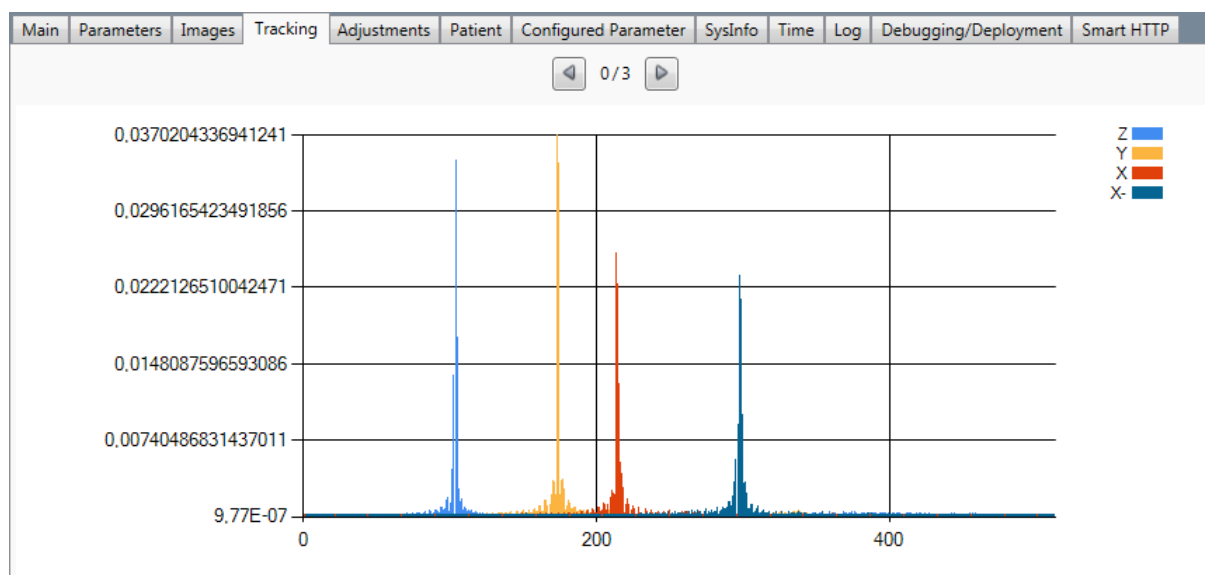


Figure 36: Tracking data shown in Access-i Test Client.

3.3.11 Patient Service

Patient-related information is shown in the “Patient” Tab:

| Main | Parameters | Images | Tracking | Adjustments | Patient | Configured Parameter | Sy |
|---------------------------------|------------|--|----------|-------------|---------|----------------------|----|
| Name | | Claire Grube | | | | | |
| Patient Id | | 17.04.18-16:36:19-DST-1.3.12.2.1107.5.2.18.91587 | | | | | |
| Sex | | female | | | | | |
| Date of Birth | | 1978-11-12 | | | | | |
| Dicom Study Id | | 4 | | | | | |
| Dicom Study Instance UID | | 1.3.12.2.1107.5.2.18.91587.30000017042005574258800000022 | | | | | |
| Age | | 39 | | | | | |
| Height | | 1.81 | | | | | |
| Weight | | 60 | | | | | |
| Direction | | headFirst | | | | | |
| Position | | supine | | | | | |
| Language | | FR | | | | | |

Figure 37: Patient service in Access-i Test Client.

3.3.12 Issue Service

Issues represent popups that can arise on the MR Host, e.g. “Automatic Table Move”, “SAR Limits Exceeded” and others. If an issue comes up, it is marshalled to the client via the Issue Service / corresponding webSocket message. An issue is marshalled only to the client that has control over the MR Host.

You can enforce a popup from the Access-i Simulator Issue Trigger Tab:



Figure 38: Triggering the Stimulation Warning Issue in the Access-i Simulator

If this is done, the following “dummy” popup comes up in the Access-i Test Client:

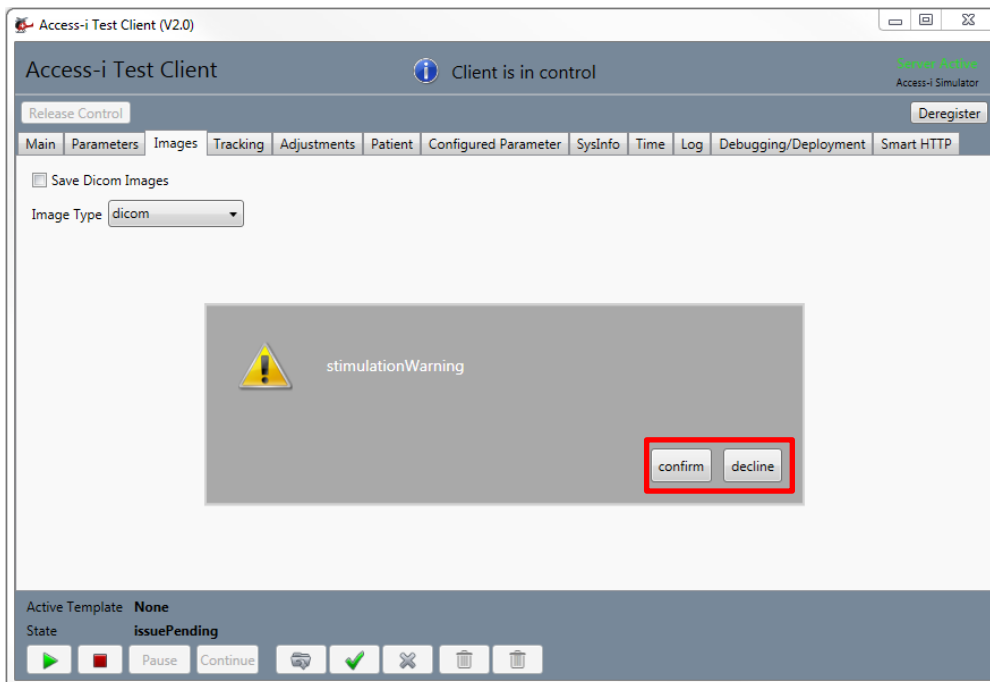


Figure 39: Confirming or Declining Stimulation Warning Issue in the Access-i Test Client

You need to either confirm or decline the issue before triggering any other operation.

3.3.13 Configured Parameter

The supported configured parameters of the preview tab are shown on the test client's dedicated tab.

The screenshot shows the 'Configured Parameter' tab with the following parameters:

| Parameter | Value | Unit |
|-----------------|-------|------|
| Slices | 1 | |
| Slice thickness | 2 | mm |
| FoV read | 300 | mm |
| FoV phase | 100 | % |
| TR | 130 | ms |
| Multiple Series | Off | |

Figure 40: Configured parameter service on test client.

3.3.14 System Information

All information that can be retrieved by the system information service is shown on a dedicated tab.

The screenshot shows the 'SysInfo' tab with the following system information:

| Parameter | Value |
|--------------------|---|
| Isocenter Position | undefined |
| Handball State | idle |
| System Information | System Model: Skyra Field Strength: 3.0 Gradient Type: XQ Number of receive channels: 24 Software version: N4_VE11C_LATEST_20160120 |

Figure 41: System information service on test client.

3.3.15 Time Service

All information that can be retrieved by the time service is shown on a dedicated tab.

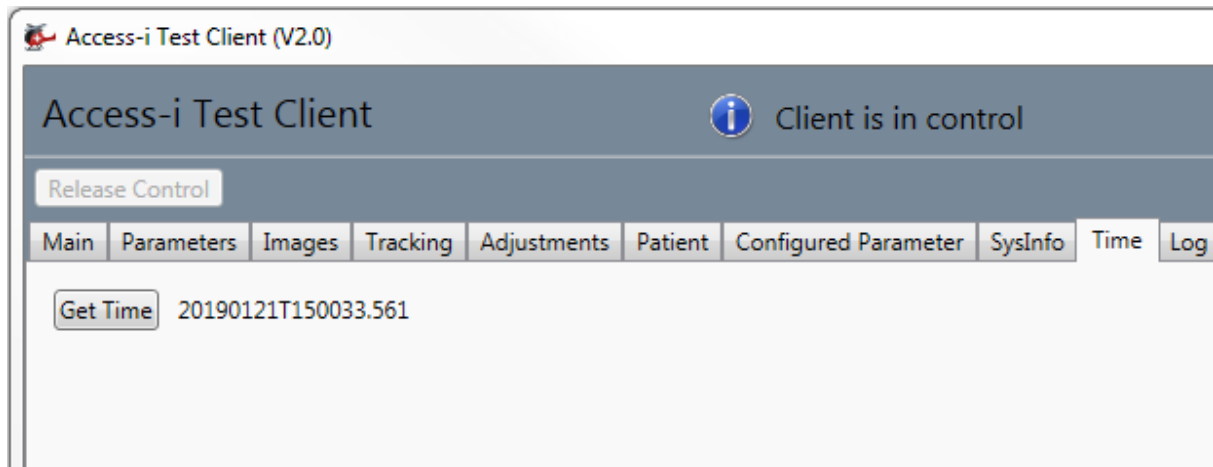
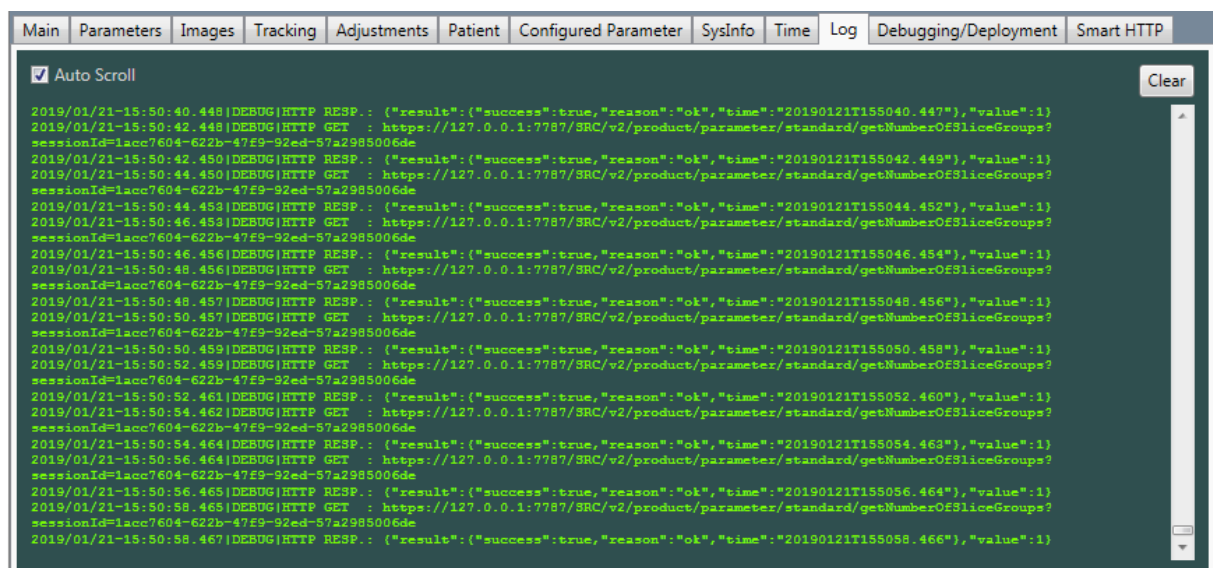


Figure 42: Time service in test client.

3.3.16 Log Window

The log window logs all requests and responses of the Test Client. Additionally, potential errors are logged here:



3.3.17 Smart HTTP

Arbitrary https GET and POST commands can be triggered by the test client. Using the Smart HTTP user interface you just need to select a service and a service method, and the complete URL is created, including session Id and operation parameters. The user just needs to fill potential parameter values. The JSON response is printed out in the text field below. This mechanism can be used to test any functionality which is not directly covered by the Access-i Test Client or for support the development of other test clients.

| Main | Parameters | Images | Tracking | Adjustments | Patient | Configured Parameter | SysInfo | Time | Log | Debugging/Deployment | Smart HTTP |
|--|------------|--------|----------|-------------|---------|----------------------|---------|------|-----|----------------------|------------|
| <div> <div>Service: remote</div> <div>Method: GetVersion</div> </div> <div> <div>URL: https://127.0.0.1:7787/SRC/v2/product/remote/getVersion?sessionId=1acc7604-622b-47f9-92ed-57a2985006de</div> <div>Submit</div> </div> <div> <div>Response: {"result":{"success":true,"reason":"ok","time":"20190121T155445.719"},"value":"2.0"}</div> </div> | | | | | | | | | | | |

Figure 43: Example for GET HTTPS command in the Test Client's Smart HTTP UI

| Main | Parameters | Images | Tracking | Adjustments | Patient | Configured Parameter | SysInfo | Time | Log | Debugging/Deployment | Smart HTTP |
|---|------------|--------|----------|-------------|---------|----------------------|---------|------|-----|----------------------|------------|
| <div> <div>Service: parameter/standard</div> <div>Method: SetSliceThickness</div> </div> <div> <div>URL: https://127.0.0.1:7787/SRC/v2/product/parameter/standard/setSliceThickness</div> <div>Submit</div> </div> <div> <div>Params: {"sessionId":"1acc7604-622b-47f9-92ed-57a2985006de","value":1.0,"allowSideEffects":true}</div> </div> <div> <div>Response: {"result":{"success":true,"reason":"ok","time":"20190121T155538.514"},"valueSet":1}</div> </div> | | | | | | | | | | | |

Figure 44: Example for POST HTTPS command in the Test Client's Smart HTTP UI

4 Annex

4.1 Recommended Libraries for Remote Clients

Based on broad experience in implementing various test clients in different programming languages the following software libraries in Table 3 are recommended for RCs to communicate with Access-i.

| Programming Language | HTTPS REST | WebSocket |
|----------------------|---|---|
| C# | .NET base functionality https://msdn.microsoft.com/en-us/library/jj819168.aspx | websocket-sharp https://github.com/sta/websocket-sharp |
| C++ | Boost http://www.boost.org/ | websocketpp https://github.com/zaphoyd/websocketpp |
| Java | Base Java (java.net.URL) https://docs.oracle.com/javase/7/docs/api/java/net/URL.html | nv-websocket https://github.com/TakahikoKawasaki/nv-websocket-client |
| Python | Python Requests http://docs.python-requests.org/en/master/ | websockets https://pypi.python.org/pypi/websockets |

Table 3: Recommended software libraries for RC implementation.

4.2 Configuration of the Configurable Parameter Card

The following description explains how to configure the configurable parameter card within the AddIn's configuration as prerequisite for using the configured parameters remotely.

In the Program Editor of the Dot Cockpit, open a template program step in Edit Mode:

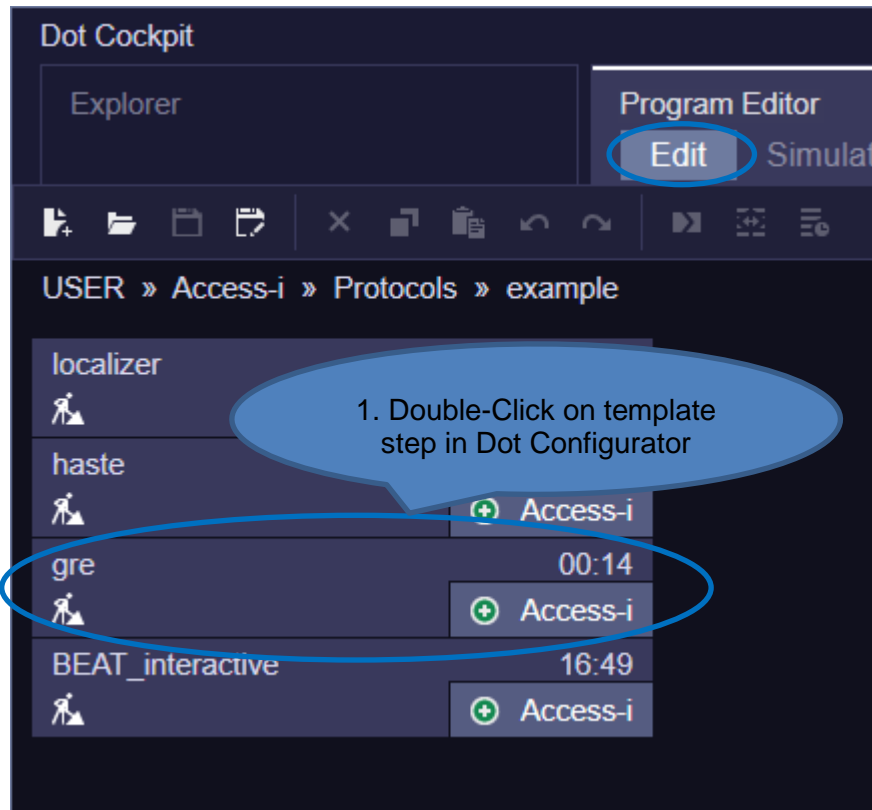


Figure 45: Template Program in Dot Cockpit

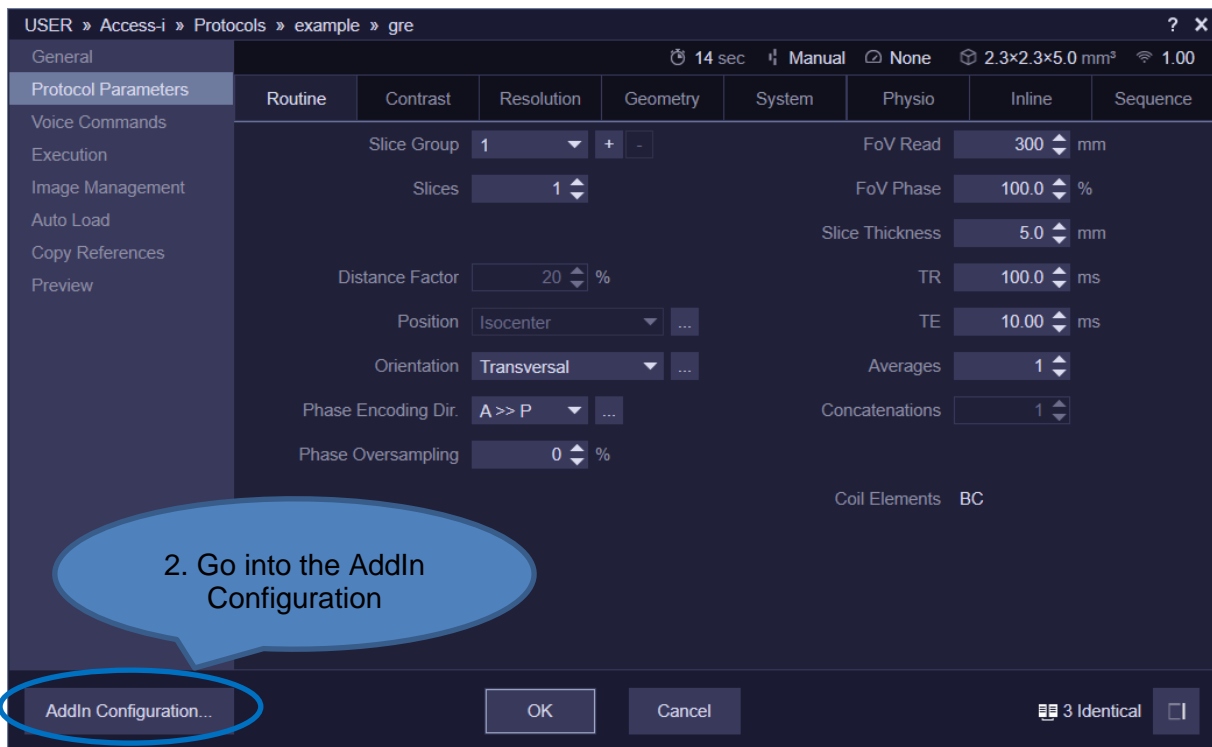


Figure 46: Step Properties Dialog

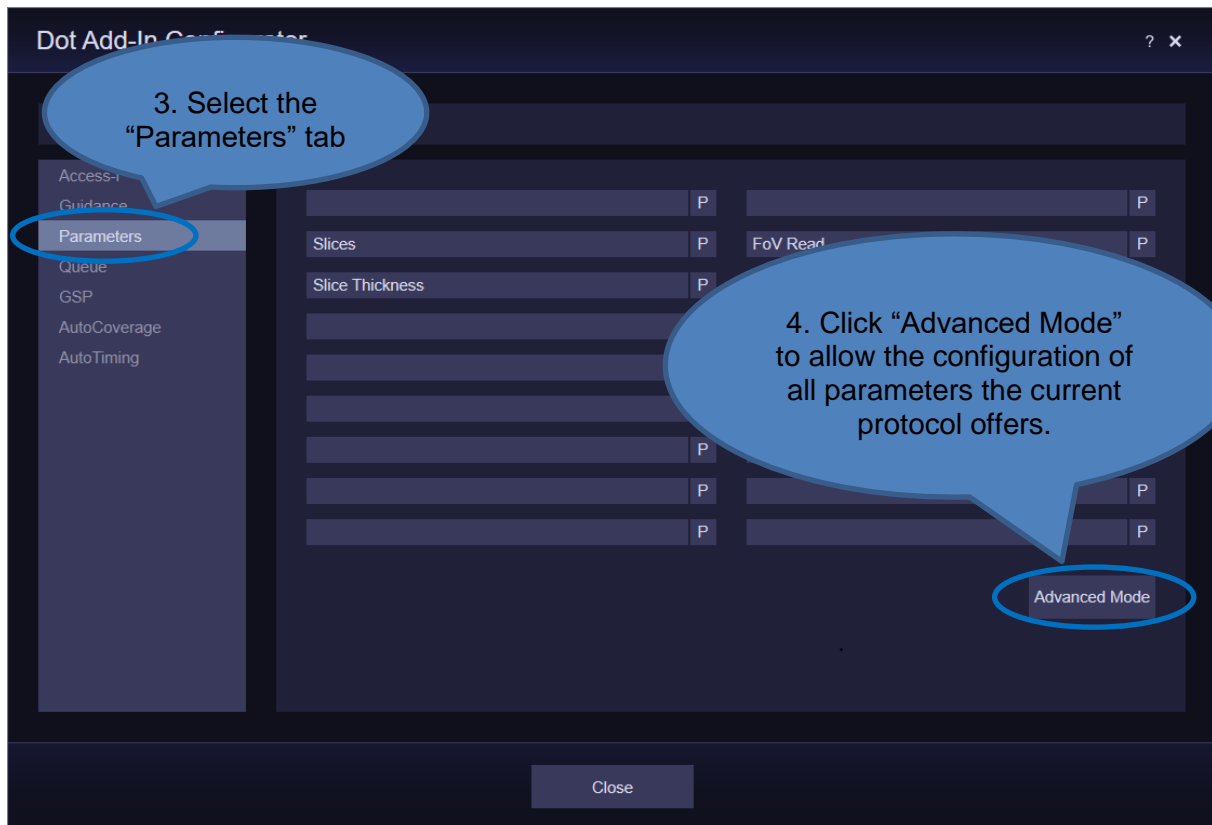


Figure 47: Parameters Tab in the AddIn Configuration

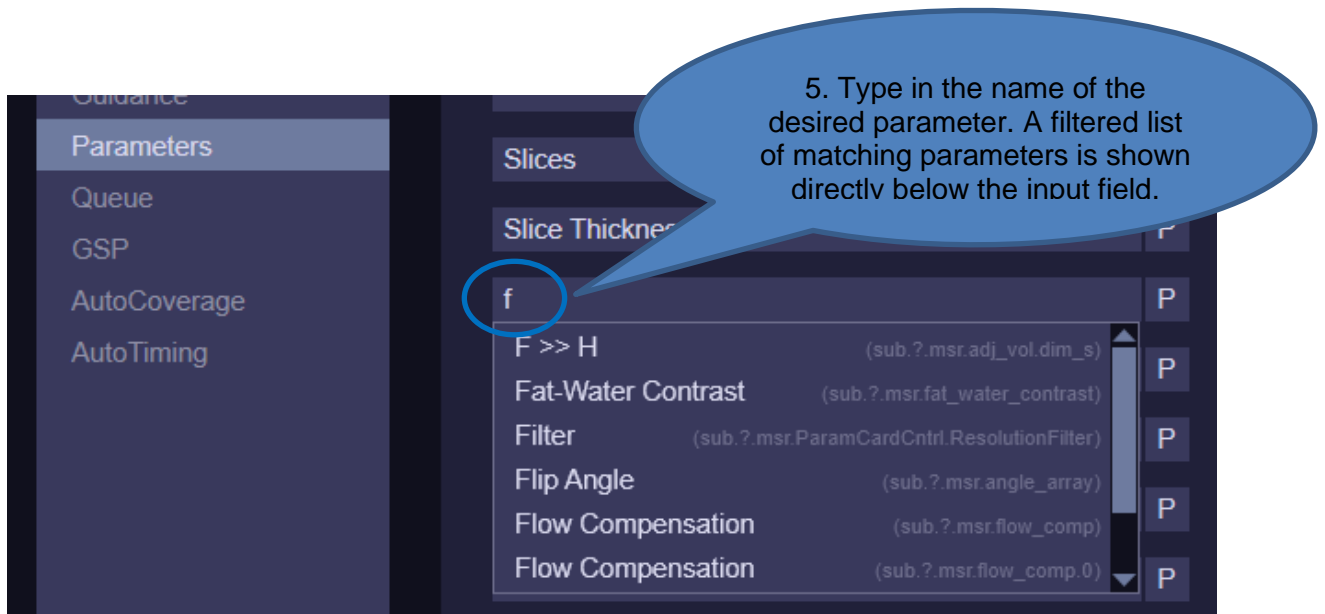


Figure 48: Configuring protocol parameter

You can do this for up to 18 entries within the configuration UI of the configurable parameter card ("Parameters"), e.g. as follows:



Figure 49: Configured Parameters Card

The final result of the configuration process can be seen on the 'Preview' tab of the "Step Properties" dialog:

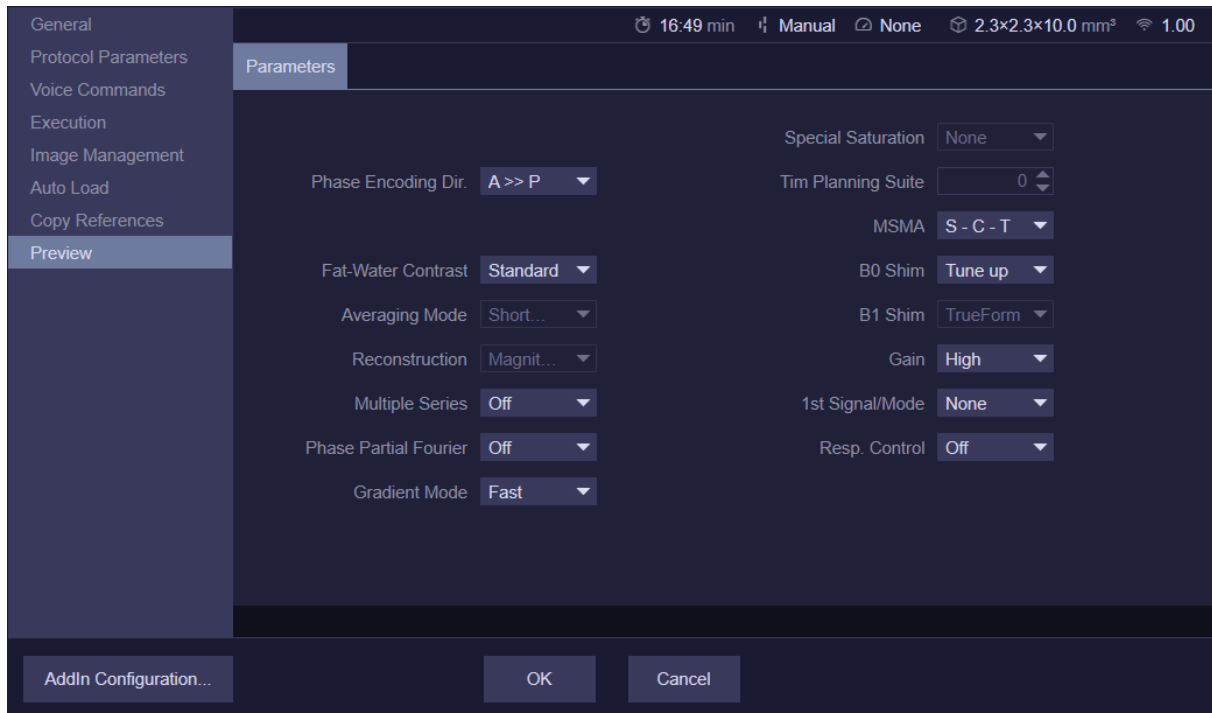


Figure 50: Parameter Preview

4.3 OpenSource Licenses

The SRC Test server and client are utilizing various open source libraries. The respective licenses can be found below.

4.3.1 Websocket-sharp

The MIT License (MIT)

Copyright (c) 2010-2017 sta.blockhead

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

4.3.2 Fleck

The MIT License

Copyright (c) 2010-2016 Jason Staten

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

4.3.3 Fellow Oak DICOM

Fellow Oak DICOM

Copyright (c) 2012-2017 fo-dicom contributors

This software is licensed under the Microsoft Public License (MS-PL)

Microsoft Public License (MS-PL)

This license governs use of the accompanying software. If you use the software, you accept this license. If you do not accept the license, do not use the software.

1. Definitions

The terms "reproduce," "reproduction," "derivative works," and "distribution" have the same meaning here as under U.S. copyright law. A "contribution" is the original software, or any additions or changes to the software. A "contributor" is any person that distributes its contribution under this license. "Licensed patents" are a contributor's patent claims that read directly on its contribution.

2. Grant of Rights

(A) Copyright Grant- Subject to the terms of this license, including the license conditions and limitations in section 3, each contributor grants you a non-exclusive, worldwide,

royalty-free copyright license to reproduce its contribution, prepare derivative works of its contribution, and distribute its contribution or any derivative works that you create.

(B) Patent Grant- Subject to the terms of this license, including the license conditions and limitations in section 3, each contributor grants you a non-exclusive, worldwide, royalty-free

license under its licensed patents to make, have made, use, sell, offer for sale, import,

and/or otherwise dispose of its contribution in the software or derivative works of the contribution in the software.

3. Conditions and Limitations

(A) No Trademark License- This license does not grant you rights to use any contributors' name,

logo, or trademarks.

(B) If you bring a patent claim against any contributor over patents that you claim are infringed

by the software, your patent license from such contributor to the software ends automatically.

(C) If you distribute any portion of the software, you must retain all copyright, patent, trademark,

and attribution notices that are present in the software.

(D) If you distribute any portion of the software in source code form, you may do so only under this

license by including a complete copy of this license with your distribution. If you distribute

any portion of the software in compiled or object code form, you may only do so under a license

that complies with this license.

(E) The software is licensed "as-is." You bear the risk of using it. The contributors give no express

warranties, guarantees or conditions. You may have additional consumer rights under your local

laws which this license cannot change. To the extent permitted under your local laws, the

contributors exclude the implied warranties of merchantability, fitness for a particular purpose

and non-infringement.

---- libijg (from DCMTK 3.5.4 COPYRIGHT) ----

Unless otherwise specified, the DCMTK software package has the following copyright:

/*

* Copyright (C) 1994-2004, OFFIS

*

* This software and supporting documentation were developed by

*

* Kuratorium OFFIS e.V.

* Healthcare Information and Communication Systems

* Escherweg 2

* D-26121 Oldenburg, Germany

*

* THIS SOFTWARE IS MADE AVAILABLE, AS IS, AND OFFIS MAKES NO WARRANTY
* REGARDING THE SOFTWARE, ITS PERFORMANCE, ITS MERCHANTABILITY OR
* FITNESS FOR ANY PARTICULAR USE, FREEDOM FROM ANY COMPUTER
DISEASES OR
* ITS CONFORMITY TO ANY SPECIFICATION. THE ENTIRE RISK AS TO QUALITY AND
* PERFORMANCE OF THE SOFTWARE IS WITH THE USER.

*

* Copyright of the software and supporting documentation is, unless
* otherwise stated, owned by OFFIS, and free access is hereby granted as
* a license to use this software, copy this software and prepare
* derivative works based upon this software. However, any distribution
* of this software source code or supporting documentation or derivative
* works (source code and supporting documentation) must include the
* three paragraphs of this copyright notice.

*

*/

The dcmjpeg sub-package includes an adapted version of the Independent JPEG
Group Toolkit Version 6b, which is contained in dcmjpeg/libijg8,
dcmjpeg/libijg12 and dcmjpeg/libijg16. This toolkit is covered by the
following copyright. The original README file for the Independent JPEG
Group Toolkit is located in dcmjpeg/docs/ijg_readme.txt.

/*

* The authors make NO WARRANTY or representation, either express or implied,
* with respect to this software, its quality, accuracy, merchantability, or
* fitness for a particular purpose. This software is provided "AS IS", and you,
* its user, assume the entire risk as to its quality and accuracy.

*

* This software is copyright (C) 1991-1998, Thomas G. Lane.
* All Rights Reserved except as specified below.

*

* Permission is hereby granted to use, copy, modify, and distribute this
* software (or portions thereof) for any purpose, without fee, subject to these
* conditions:

* (1) If any part of the source code for this software is distributed, then this
* README file must be included, with this copyright and no-warranty notice
* unaltered; and any additions, deletions, or changes to the original files

- * must be clearly indicated in accompanying documentation.
- * (2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".
- * (3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.
- *
- * These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.
- *
- * Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".
- *
- * We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.
- */

---- OpenJPEG JPEG 2000 codec (from license.txt) ----

/*

- * Copyright (c) 2002-2007, Communications and Remote Sensing Laboratory, Universite catholique de Louvain (UCL), Belgium
- * Copyright (c) 2002-2007, Professor Benoit Macq
- * Copyright (c) 2001-2003, David Janssens
- * Copyright (c) 2002-2003, Yannick Verschueren
- * Copyright (c) 2003-2007, Francois-Olivier Devaux and Antonin Descampe
- * Copyright (c) 2005, Herve Drolon, FreeImage Team
- * All rights reserved.
- *
- * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions

- * are met:
- * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- *
- * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS `AS IS'
- * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
- * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
- * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
- * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
- * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
- * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
- * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
- * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
- * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
- * POSSIBILITY OF SUCH DAMAGE.
- */

---- CharLS JPEG-LS codec (from License.txt) ----

Copyright (c) 2007-2009, Jan de Vaan
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice,

this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of my employer, nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"

AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR

ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON

ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS

SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---- Unity.IO.Compression (from LICENSE.TXT and PATENTS.TXT) ----

The MIT License (MIT)

Copyright (c) Microsoft Corporation

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Microsoft Patent Promise for .NET Libraries and Runtime Components

Microsoft Corporation and its affiliates ("Microsoft") promise not to assert any .NET Patents against you for making, using, selling, offering for sale, importing, or distributing Covered Code, as part of either a .NET Runtime or as part of any application designed to run on a .NET Runtime.

If you file, maintain, or voluntarily participate in any claim in a lawsuit alleging direct or contributory patent infringement by any Covered Code, or inducement of patent infringement by any Covered Code, then your rights under this promise will automatically terminate.

This promise is not an assurance that (i) any .NET Patents are valid or enforceable, or (ii) Covered Code does not infringe patents or other intellectual property rights of any third party. No rights except those expressly stated in this promise are granted, waived, or received by Microsoft, whether by implication, exhaustion, estoppel, or otherwise.

This is a personal promise directly from Microsoft to you, and you agree as a condition of benefiting from it that no Microsoft rights are received from suppliers, distributors, or otherwise from any other person in connection with this promise.

Definitions:

"Covered Code" means those Microsoft .NET libraries and runtime components as made available by Microsoft at <https://github.com/Microsoft/referencesource>.

".NET Patents" are those patent claims, both currently owned by Microsoft and acquired in the future, that are necessarily infringed by Covered Code. .NET Patents do not include any patent claims that are infringed by any Enabling Technology, that are infringed only as a consequence of modification of Covered Code, or that are infringed only by the combination of Covered Code with third party code.

".NET Runtime" means any compliant implementation in software of (a) all of the required parts of the mandatory provisions of Standard ECMA-335 – Common Language Infrastructure (CLI); and (b) if implemented, any additional functionality in Microsoft's .NET Framework, as described in Microsoft's API documentation on its MSDN website. For example, .NET Runtimes include Microsoft's .NET Framework and those portions of the Mono Project compliant with (a) and (b).

"Enabling Technology" means underlying or enabling technology that may be used, combined, or distributed in connection with Microsoft's .NET Framework or other .NET Runtimes, such as hardware, operating systems, and applications that run on .NET Framework or other .NET Runtimes.

4.4 Changelog

| Version Date | Author, Dept. | Reason |
|--------------------|-------------------------------|---|
| V1.1.1, 2019-10-28 | Dirk Franger SHS DI MR AST | Type of Patient “dicomStudyId” changed from int - > string in Numaris X systems. |
| V1.1.2, 2021-06-28 | Dirk Franger | Added systemInformation/getSerialNumber which was introduced with VA30A Fixed typo “pixelRepresentation” |

This document has been created using template 8388170-ELH-DOT-07.

Copyright © Siemens Healthcare GmbH. All rights reserved. Classification „Restricted“.