Protocol Implementation and Definition Guide
R12.1

File: Protocol Definition Guide.doc
Version: 1.03
Date: 6th September 2007

Contents

1. Document History

1.1. Revision History

Version	Date	Comments	Author
1.00	18/09/2006	Updated for R12.0.	K. Currie
1.01	20/09/2006	Added spellCheck attribute to text, textArea and column elements.	K. Currie
1.02	11/01/2007	Added patient_sex mandatory field notes	S. Thomson
1.03	06/09/07	Added new pre-population items, and new questions/column ids allowed to Appendix A, B & C	S. Thomson

1.2. Distribution

This document has been distributed to

Name	Title	Date of Issue	Version
Gateway Development Team		06/09/2007	1.03

2. Introduction

2.1. Overview

The SCI Gateway product enables electronic communication of clinical data between organisations within the NHS Scotland. These communications take the form of XML (eXtended Mark-up Language) messages compliant with published NHS XML schemas.

Currently six such schemas have been established.

“referral.xsd” – for Referral messages
“discharge.xsd” - for Discharge messages
“general.xsd” – used by both the referral and discharge schemas
“eGPFR-Request.xsd” – for eGPFR request messages
“eGPFR-Response.xsd” – for eGPFR response messages
“eGPFR.xsd” – used by both the request and response schemas

A method that the SCI Gateway implements to create these clinical messages is known as “protocol-based referrals”, but can be applied equally well to other communications such as discharge messages or responses. This document provides a guide to authoring protocols.

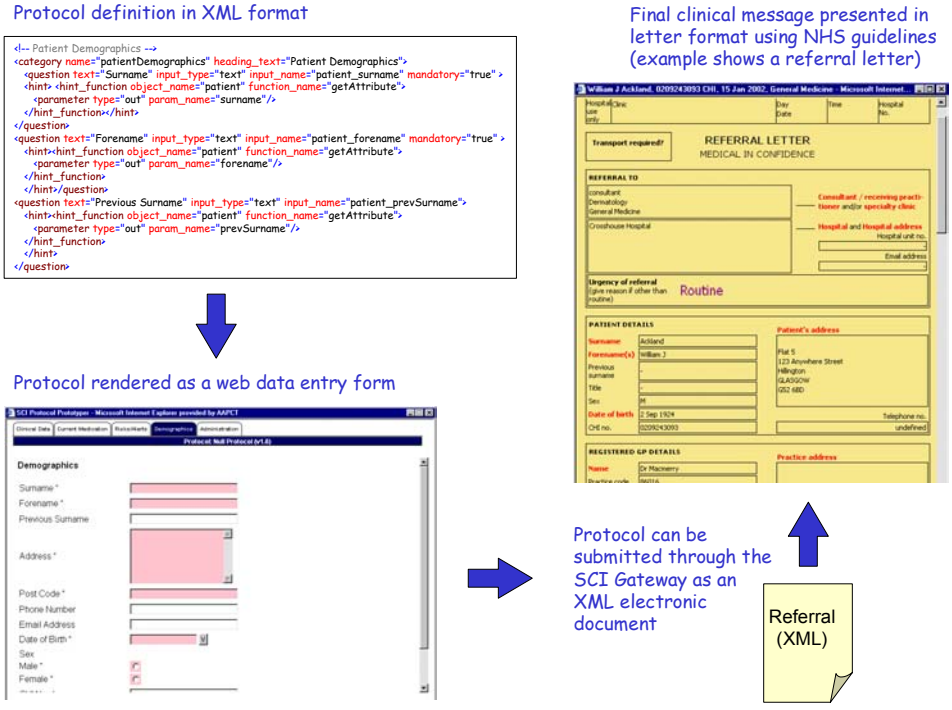


Figure 1 - How protocols are used by the SCI Gateway

A “protocol” is the definition of the data requirement and structure for these clinical communications. With SCI, protocols are defined as XML documents that are “transformed” to web data entry forms (see Figure 1
Figure 1). The protocol definition contains a set of questions that capture the demographic and clinical data required for the communication. The SCI Gateway transforms the XML into an HTML data entry form.

2.2. Additional References

Document	Version	Description

3. Protocol Implementation

3.1. Getting Started

In the first instance it is important to ensure that any perspective System Administrator / Protocol writer (these roles may be performed by the same resource and this document will refer to the roles as system administrator) have experience in producing XML documents. Although this guide is extensive on describing the required XML content of Protocols a certain amount of intuitiveness is still required.

The following outlines the required steps to initiate the protocol implementation process.

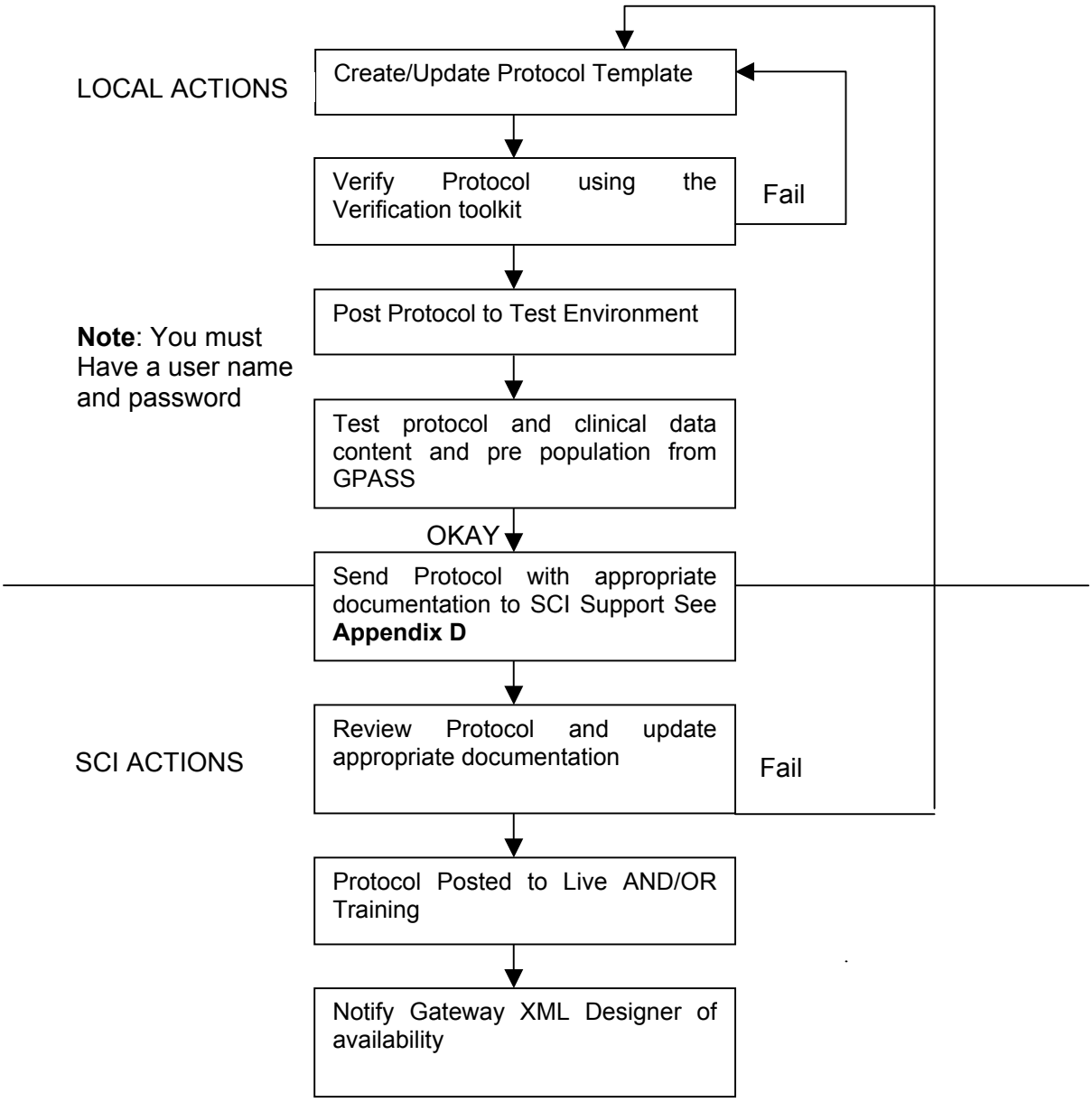
- Identify a system administrator
- Inform SCI Support for the system administrator to be added as a user to the SCI Gateway Test environment.
- Contact your Implementation Manager to arrange training in the use of the administration of Protocols within the Gateway environment.
- Ensure that you have the most recent version of the “Protocol Verification Tool”. Copies may be obtained by contacting the SCISupport@gpass.csa.scot.nhs.uk or downloaded from the web site http://www.show.scot.nhs.uk/sci/products/gateway/gate_down.htm

3.2. Naming Conventions

Name, version and region information for the protocol is described in section “Element Definitions”.

3.3. The Process

The following flow chart defines the process:



Note: Protocols tested for current release will require a user name and password for live Gateway Server. Protocols to be tested for future releases will require a user name and password for the test Gateway Server. These can be obtained by emailing: SCISupport@gpass.csa.scot.nhs.uk

3.3.1. Create Protocol

Start basic and build upon what works (there is a copy of a Superset Protocol included in the CD which can be used for reference and editing), ensure that appropriate clinical information is included as required by providers (bells and whistles can be added later) and always consider 'clinical risk' as it applies to your GP's and hospital providers.

The Protocol Verification Tool supports a variety of editors including word pad. However, it is recommended that the Trust use their preferred XML editor, which the trust will be responsible for buying and training their staff to use. The typical method of developing a protocol will be to cut and paste from existing Superset protocol.

3.3.2. Verify Protocol

The Protocol Verification Tool tests the validity of the XML protocol produced, to view the screens as the end user would, to enter information as per end users, to view the print format of the referrals, and to see the referral xml created from the screens.

The skills needed to use the tool are intermediate windows skills, knowledge of XML, experience of HTML and some exposure to basic programming. The tool is intended for technical staff from Trusts working with end users.

3.3.3. Post Protocol to Test Environment

In order to perform the end-to-end testing of the protocol, it must be 'posted' to the test environment as follows:

- Select the instance 'Test' and logon to Gateway
- Select the Admin menu item
- Click on Protocols
- When the Protocol screen is viewed click on Add
- In the name field enter the name you wish to call the protocol
- Give it a brief description
- Associate the protocol with the correct Message Type
- Click on Protocol HCEs and select the area which you want to have access to the protocol i.e. GP Practices and Hospitals
- Select the appropriate hospital and specialty – each specialty has associated consultants and tick the appropriate box and choose Select
- To add the XML click on the Protocol XML button which will display the form
- Copy and paste the XML into the box and Close
- Click on the Active box to enable the protocol
- Click OK when finished

3.3.4. Test Protocol

This is the most significant part of the protocol process. The testing must be rigorous and comprehensive and tested using GPASS or a third party GP system to ensure the correct fields in the protocol are being populated as defined in the XML. The most common problems with any implementation can be traced back to insufficient testing.

3.3.5. Send Protocol and Supporting Documentation to SCI

SCI require all documentation supporting the protocol for review prior to 'posting' to the live and training Gateway. SCI aim to publish the new protocols and supporting documentation on the SCI Support web site with the agreement of the Trust.

Checklist:

- The protocol (XML)
- Information required for protocols (See Appendix - Protocol Specification Template)
- Agreement to publish the documents on the SCI Support web site.

The documents should be sent in electronic format to SCISupport@gpass.csa.scot.nhs.uk

3.3.6. Review of Protocol and Supporting Documentation

SCI Support will review the protocol as part of the SCI quality assurance procedures. Any issues will be discussed and actions agreed.

3.3.7. Protocol Posted to Training and Live Gateway

SCI Product Support will be responsible for posting the protocols onto the training and live instances of the Gateway. This will normally be achieved within 1 working day, however, for Service Level Agreement purposes 3 working days should be allowed for.

4. Protocol Definition

4.1. XML – The Basics

SCI protocols are written as XML documents; therefore some knowledge of XML is required to author a protocol as this document provides only the briefest introduction. Refer to the numerous books and on-line resources now available for XML to gain a more complete understanding.

XML documents are structured text files that can both contain data and describe data. This makes them much more powerful that text-based file formats such as comma separated values (CSV) or plain ASCII files.

XML documents can be created with any standard text editor, such as Notepad. However, more advanced tools such as XML Spy (http://www.xmlspy.com) are recommended as they incorporate validation functions and colour coding of elements.

4.1.1. A Simple XML Document

Below is an example of a simple XML document that could be used to describe a car.

```
<car registration=”AB51 CFP”>
        <make>Ford</make>
        <model>Mondeo</model>
        <type doors="4">Saloon</type>
        <color>Blue</color>
        <engine capacity="1998cc" cylinders="4"></engine>
</car>
```

XML uses nested elements (tags surrounded in < and > characters) and attributes within these elements to hold document data. All elements must be opened and closed. An element is opened like this `<car>` and closed like this `</car>`. The contents, or data, within the element is written between the opening and closing tags. Together with text data, elements can contains other elements and nesting elements in this way is a key feature of XML.

Attributes have a name and a value and can be added to elements. In the car example shown, the number of doors is held as an attribute of the `<type>` element using the syntax `doors=”4”`. All attribute values must be surrounded in quotes (“…”) and can appear in any order within the element definition.

4.1.2. Six Key Things to Remember When Using XML

All element tags must be closed

When an element does not contain data, but just attributes, it can be opened and closed in one statement. For example `<engine capacity="1998cc" cylinders="4"></engine>` can be shortened to `<engine capacity="1998cc" cylinders="4"/>`. Note the additional ‘/’ character to denote that the element is closed. You will see notation used within many of the protocols developed by the SCI team.

XML is case sensitive

Be careful not to mistype element and attribute names. For example `<engine capacity="1998cc"/>` and `<Engine Capacity="1998cc"/>` would be interpreted as different element types by an XML program.

All attribute values must be contained in quotes ("…")

Use double quotes when defining attribute values. For example, `<engine capacity="1998cc"/>` is valid where as `<engine capacity=1998cc/>` is invalid.

Tags must be correctly nested

Most popular web browsers will accept badly nested html tags (e.g. some text). When using XML the parser will require tags to be opened and closed in the correct order (e.g. some text).

White-space characters such as spaces, tabs and carriage returns are ignored

This allows an XML document to be formatted and make easier to read by the human eye. It is standard practice to add a carriage return after each element definition and indent sub-elements to make the document more readable.
Note. Spaces are not ignored when part of the data inside the XML, for example `<engine capacity="1998 cc"/>`.

Be careful using some characters within protocol definitions

Some limitations of HTML apply to XML, especially where "escape characters" are concerned. (A useful reference is http://www.internet-tips.net/HTML/Escape.htm).

Character	XML representation	What it is
<	<	Less than
>	>	More than
&	&	Ampersand
"	"	Double quote
Space		Non-breaking space

For example, if you want to encode "< than 5 cigarettes a day" into a <question> you would format your question as follows:

```
<question text="Smoking" type="select" id="riskSmoking">
        <option text="Ex-Trivial Smoker (&lt;5 cig/day)" value="Ex-Trivial smoker" id="riskSmoking_Ex-Trivial smoker"/>
</question>
```

4.1.3. A Simple Protocol XML Document

An XML document to define a protocol must follow a very specific format to enable the software to interpret the protocol and dynamically create a web-form. A very simple protocol definition is provided below.

```
<protocol name="Simple Protocol" version="1.0">
        <category name="tab1" text="My First Tab">
                        <question text="Patient Name" type="text" id="patient_name"/>
        </category>
</protocol>
```

All protocols must have a single top-level element called “protocol”. All other elements must be contained within the `<protocol>…</protocol>` element tags. The example given produces a protocol called “Simple Protocol” that asks a single question, the patient’s name. When hooked into either the SCI Protocol Verification Tool, or the main SCI Gateway application, the web-form for this protocol will look like Figure 2.

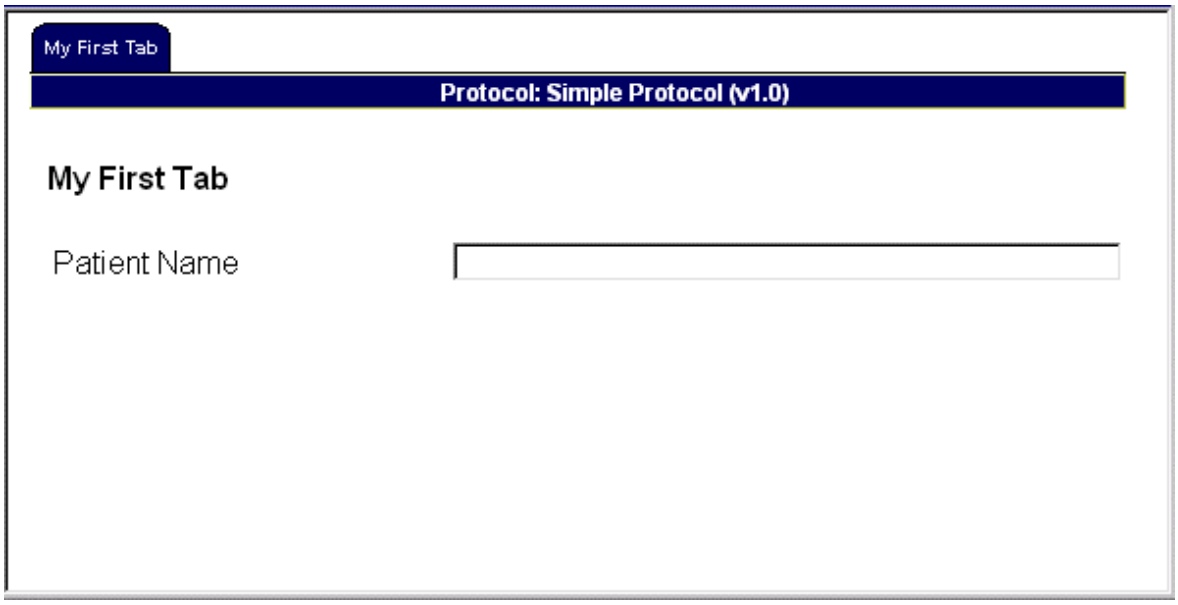


Figure 2 - Web-form for the "Simple Protocol"

The `<category>` element is presented as a tab with the heading text “My First Tab”, defined by the attribute `text`. Within this category is a single `<question>` element for the patient’s name, which should be entered in a text box. The attribute `text` defines the label for the question and the attribute `type` defines that a text box should be used. The purpose of the other attributes used in the example is covered in a later chapter.

A complete protocol comprises of a number of `<question>` elements organised into `<categories>`. More complex elements and attributes are also used to define more complex aspects of a protocol, including how responses to questions can be pre-populated by GPASS or a third party GP system used within primary care.

4.2. Element Definitions

The XML elements used within SCI protocols are described in this section.

4.2.1. Element “protocol”

The `<protocol>` element is the top-level element for the whole protocol definition and defines the name, version and region (if applicable) for the protocol. All other elements are nested inside this element.

4.2.1.1. ATTRIBUTE DEFINITIONS FOR ELEMENT “PROTOCOL”

Attribute	Usage	Description
name	Mandatory (lower case)	The name of the protocol, displayed in the header bar on the web-form together with the protocol version.
region	Optional (lower case)	An optional attribute to identify the region or owner of the protocol. This is not currently displayed to the user within the web-form but is used to distinguish similar protocols used in different regions. For example, to identify the colorectal cancer protocol for Argyll & Clyde from the colorectal cancer protocol written by Forth Valley.
version	Mandatory (lower case)	The version of the protocol, displayed in the header bar on the web-form together with the protocol name.

4.2.2. Element “category”

Used to logically separate sections of the protocol. Categories can be nested (i.e. categories that contain categories) which affects how the category is displayed within the web-form for the protocol.

4.2.2.1. TOP-LEVEL CATEGORIES

These are displayed as tabs along the top of the web-form and repeated as a main heading at the top of each tab area. The number of top-level categories allowed within a protocol is restricted by the width of the protocol on-screen. If too many categories are added, the protocol web-form will be displayed with scroll-bars. If many categories are required, use short titles (e.g. rename “Patient Demographics” to just “Demographics”) to reduce the size of the tab on-screen.

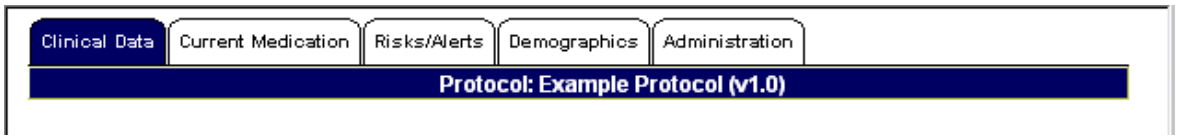


Figure 3 - Using top-level categories

The example shown in Figure 3 contains five top-level categories. An extract of the XML definition for this protocol is provided below. Explanations for the meaning of the attributes used with the `<category>` are provided at the end of this section.

```
<protocol name="Example Protocol" version="1.0">
        <category text="Clinical Data">
        ...
        </category>
        <category text="Current Medication">
        ...
        </category>
        <category text="Risks/Alerts">
        ...
        </category>
        <category text="Demographics">
        ...
        </category>
        <category text="Administration">
        ...
        </category>
</protocol>
```

4.2.2.2. SUB-LEVEL CATEGORIES

Separate sections can be added to a top-level category (a single tab in the protocol) by using additional category elements. A second-level category is displayed as a sub-title using a smaller font size than the main title. Additional sub-categories will use a smaller font-size again.

Figure 4 shows an example of using three levels of category within a protocol. The “Clinical Data” tab is a top-level category containing two second-level categories (“Main Condition” and “Priority”). Within “Main Condition” is a third-level category “Additional Information”.

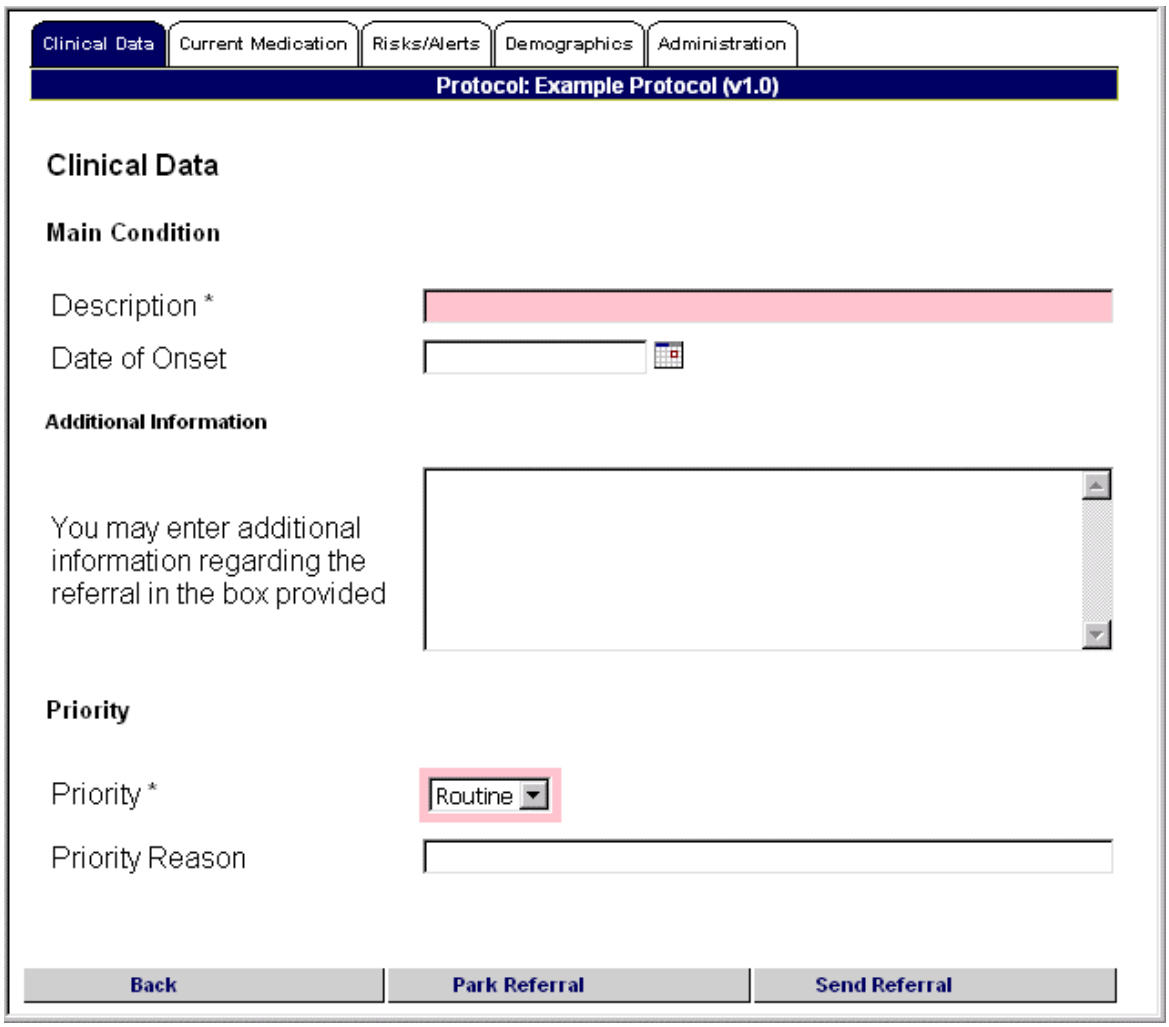


Figure 4 - Using "Categories"

Each category contains `<question>` elements that define the data-entry parts of a protocol.

4.2.2.3. ATTRIBUTE DEFINITIONS FOR ELEMENT “CATEGORY”

Attribute	Usage	Description
level	Obsolete (lower case)	Previous implementations of the SCI Gateway used a “level” attribute. This attribute is now obsolete and ignored.
text	Mandatory (lower case)	The heading text for the category displayed to the user on the web-form.

4.2.3. Element “question”

Questions form the data-entry part of a protocol and a `<question>` element defines a single question.

4.2.3.1. ATTRIBUTE DEFINITIONS FOR ELEMENT “QUESTION”

Attribute	Usage	Description
Id	Mandatory (lower case)	Each question must be given a unique name within a protocol. This name is not displayed to the user, but is required by the software within the main SCI Gateway application. See section Naming Questions Using the “id”

		Attribute.
mandatory	Optional (lower case)	Indicates that the question is mandatory and the user cannot submit the protocol until it is completed.
readonly	Optional (lower case)	Indicates that the question is read only and cannot be edited directly by the user. The read only state for a question is only activated if the question has been pre-populated by a 3rd party application (i.e. GPASS). It is most applicable for patient identification data such as CHI.
text	Mandatory (lower case)	The label for the question displayed to the user on the web-form.
type	Mandatory (lower case)	The type of question. See section Question Types.
userName	Optional (camel case)	A friendly name for the question. This is only ever used in conjunction with a mandatory="true" attribute and is used in the error message to inform the user that the mandatory field has not been filled in.
validate_type	Obsolete (lower case)	Used in an earlier implementation of the SCI Gateway. This attribute is now obsolete and ignored.

4.2.4. Naming Questions Using the "id" Attribute

The "id" attribute of the `<question>` element is a key field that enables the protocol to be successfully processed by the SCI Gateway.

It provides each question with a unique ID that is used to map the field to the appropriate location within the NHS XML schema that defines the physical format of the clinical communication.

4.2.4.1. PRE-DEFINED STANDARD FIELDS

There are standard "named fields" that the Gateway is programmed to recognise and should be identical across **all** protocols (refer to Appendix - Predefined "id" Values - for a full list).

If the naming standards are ignored, then the Gateway will not be able to create a full clinical communication message (e.g. a referral). At best, the message will not be accepted by the Gateway. At worst, clinical risk may be introduced.

Even if naming standards are correctly followed, the protocol XML has to be properly arranged and verified manually. The protocol verification tool can tell you what standard fields are missing from a protocol, but it cannot tell you if the protocol is semantically correct. For example, if a named field is in the wrong place in the protocol XML, but adheres to the naming standard the protocol verification tool will not detect the error.

The XML below contains two errors:

```
<question text="Forename" id="patient_surname" type="text" mandatory="true"
userName="Patient Surname">
</question>

<question text="Surname " id="patient_forename" type="text" mandatory="true"
userName="Patient Forename">
</question>
```

If the protocol is uploaded to the Gateway and then used, the patient's forename will be recorded as surname, and vice versa, in the clinical communication message.

Use the protocol verification tool's **Preview Letter** functionality to detect and correct such errors.

4.2.4.2. PROTOCOL-SPECIFIC FIELDS

Questions relating to protocol-specific data can be added into a protocol. Such data is classified as either an administrative, investigation or examination question.

To store custom data within the Gateway, the "id" attribute **must** have one of the following prefixes:

"admin_" + unique name (to denote an administrative question)

"investigation_" + unique name (to denote a clinical investigation question)

"examination_" + unique name (to denote a clinical examination question)

Administrative questions are normally used to provide additional information to admin or medical records staff to aid them when dealing with the patient. In the case of referrals, most are handled first by medical records then passed onto consultants. Some referral protocols are now including questions relating to patient hearing or sight deficiencies that will help staff provide better care to the patient. When a referral letter is presented in the SIGN Guidelines format information captured as administrative questions are presented on the first page together with the patient demographic data.

Investigation and **examination** questions are for capturing clinical data required by the consultant. In the case of referrals, such data is presented on the second page of the SIGN Guidelines letter when printed.

IF YOU DO NOT USE THE NAMING CONVENTION ABOVE, THERE IS A RISK THAT YOUR PROTOCOL-SPECIFIC DATA WILL NOT BE RECORDED WITHIN THE GATEWAY.

4.2.4.3. QUESTION TYPES

Eight types of question are supported within SCI protocols, plus one type used for internal system processes. Each type of question results in a different type of data entry control used on the web-form. The type of question is defined by the "type" attribute of the `<question>` element.

Input Type	Description	Notes
check	Check box (lower case)	For Yes/No responses using a check box.
date	Date picker (lower case)	Displays a date picker control that allows the user to scroll between months/years and select a date.
dynaTable	Variable size table (camel case)	Arranges data entry within a tabular format that can dynamically grow as more data is required.
hidden	Hidden field (lower case)	Used specifically for internal system processes of the SCI Gateway. All protocols need to include one specific hidden question that holds the GPASS patient key to enable GPASS integration. Hidden fields are used by the system and should not be

		specified by protocol writers unless there is a good reason for it.
radio	Radio group (lower case)	To select a value from a pre-defined set of options.
select	Drop-down list (lower case)	To select a value from a pre-defined list.
table	Fixed size table (lower case)	Arranges data entry within a tabular format.
text	Single-line text box (lower case)	Allows a single line of text entry. A useful attribute for this input type is maxlength (lowercase). E.g. maxlength="50". This attribute specifies the maximum number of characters that can be accepted from the keyboard.
textArea	Multi-line text box (camel case)	Allows multiple lines of text entry up to 100,000 characters.

4.2.4.3.1 "text"

Questions defined as type='text', will be converted to a text input HTML element on the message screen.

Attribute	Usage	Description
filters	Optional (lower case)	The filter node specifies what information will be retrieved from the Clinical events collection of the patient.
help	Optional (lower case)	The help text.
hidden	Optional (lower case)	Hides the control's label if set to true.
hint	Obsolete (lower case)	Used in an earlier implementation of SCI Gateway. These nodes are now obsolete and ignored.
id	Mandatory (lower case)	Name of the HTML text field.
mandatory	Optional (lower case)	Indicates that the question is mandatory and the user cannot submit the protocol until it is completed. An asterix (*) will be added after the label to indicate the mandatory nature.
maxlength	Optional (lower case)	Sets the maximum number of characters that can be entered by the user.
object	Optional (lower case)	Generic pre-population object.
objectOld	Optional (camel case)	GPASS pre-population object.
onkeydown	Optional (lower case)	System attribute only.
property	Optional (lower case)	Generic pre-population object's property.
propertyOld	Optional (camel case)	GPASS pre-population object's property.
readonly	Optional (lower case)	Indicates that the question is read only and cannot be edited directly by the user. The read only state for a question is only activated if the question has been pre-populated by a 3rd party application (i.e. GPASS). It is most applicable for patient identification data such as CHI.
spellCheck	Optional (camel case)	Can the question be spell checked. If the question is also readonly then spell checking will be disabled.
text	Mandatory (lower case)	Label of the HTML text field on the web-form.
userName	Optional (camel	A friendly name for the question. This is only ever

	case)	used in conjunction with a mandatory="true" attribute and is used in the error message to inform the user that the mandatory field has not been filled in.
validate_type	Obsolete (lower case)	Used in an earlier implementation of the SCI Gateway. This attribute is now obsolete and ignored.
width	(lower case)	Sets the width of the HTML input element in pixels.

4.2.4.3.2 "textArea"

Questions defined as type='textArea', will be converted to a Textarea HTML element of 6 rows on the message screen.

Attribute	Usage	Description
filters	Optional (lower case)	The filter node specifies what information will be retrieved from the Clinical events collection of the patient.
help	Optional (lower case)	The help text.
hidden	Optional (lower case)	Hides the control's label if set to true.
hint	Obsolete (lower case)	Used in an earlier implementation of SCI Gateway. These nodes are now obsolete and ignored.
id	Mandatory (lower case)	Name of the HTML textarea.
mandatory	Optional (lower case)	Indicates that the question is mandatory and the user cannot submit the protocol until it is completed. An asterix (*) will be added after the label to indicate the mandatory nature.
object	Optional (lower case)	Generic pre-population object.
objectOld	Optional (camel case)	GPASS pre-population object.
onkeydown	Optional (lower case)	System attribute only.
property	Optional (lower case)	Generic pre-population object's property.
propertyOld	Optional (camel case)	GPASS pre-population object's property.
readonly	Optional (lower case)	Indicates that the question is read only and cannot be edited directly by the user. The read only state for a question is only activated if the question has been pre-populated by a 3rd party application (i.e. GPASS). It is most applicable for patient identification data such as CHI.
spellCheck	Optional (camel case)	Can the question be spell checked. If the question is also readonly then spell checking will be disabled.
text	Mandatory (lower case)	Label of the HTML textarea on the web-form.
userName	Optional (camel case)	A friendly name for the question. This is only ever used in conjunction with a mandatory="true" attribute and is used in the error message to inform the user that the mandatory field has not been filled in.
validate_type	Obsolete (lower case)	Used in an earlier implementation of the SCI Gateway. This attribute is now obsolete and ignored.

4.2.4.3.3 "select"

Questions defined as type='select', will be converted to a Select HTML element on the message screen.

Attribute	Usage	Description
filters	Optional (lower case)	The filters node specifies what information will be retrieved from the Clinical events collection of the patient.
help	Optional (lower case)	The help text.
hidden	Optional (lower case)	Hides the control's label if set to true.
hint	Obsolete (lower case)	Used in an earlier implementation of SCI Gateway. These nodes are now obsolete and ignored.
id	Mandatory (lower case)	Name of the HTML dropdown list.
mandatory	Optional (lower case)	Indicates that the question is mandatory and the user cannot submit the protocol until it is completed. An asterix (*) will be added after the label to indicate the mandatory nature.
object	Optional (lower case)	Generic pre-population object.
objectOld	Optional (camel case)	GPASS pre-population object.
option	Optional (lower case)	Option nodes are the entries in the dropdown list. At least one option node should be specified to allow the selection of at least one option.
property	Optional (lower case)	Generic pre-population object's property.
propertyOld	Optional (camel case)	GPASS pre-population object's property.
text	Mandatory (lower case)	Label of the HTML dropdown list on the web-form.
userName	Optional (camel case)	A friendly name for the question. This is only ever used in conjunction with a mandatory="true" attribute and is used in the error message to inform the user that the mandatory field has not been filled in.

4.2.4.3.4 "date"

Questions defined as type='date', will be converted to a HTML <input> element on the message screen that is read only. It will also have a date picker control to the right of it that will allow the user to select the date into the HTML input element.

Attribute	Usage	Description
help	Optional (lower case)	The help text.
hidden	Optional (lower case)	Hides the control's label if set to true.
hint	Obsolete (lower case)	Used in an earlier implementation of SCI Gateway. These nodes are now obsolete and ignored.
id	Mandatory (lower case)	Name of the HTML date field.
mandatory	Optional (lower case)	Indicates that the question is mandatory and the user cannot submit the protocol until it is completed. An asterix (*) will be added after the label to indicate the mandatory nature.
object	Optional (lower case)	Generic pre-population object.
objectOld	Optional (camel case)	GPASS pre-population object.
property	Optional (lower case)	Generic pre-population object's property.
propertyOld	Optional (camel case)	GPASS pre-population object's property.
readonly	Optional (lower case)	Indicates that the question is read only and cannot be edited directly by the user. The read only state

		for a question is only activated if the question has been pre-populated by a 3rd party application (i.e. GPASS). It is most applicable for patient identification data such as CHI.
text	Mandatory (lower case)	Label of the HTML date field on the web-form.
userName	Optional (camel case)	A friendly name for the question. This is only ever used in conjunction with a mandatory="true" attribute and is used in the error message to inform the user that the mandatory field has not been filled in.
validate_type	Obsolete (lower case)	Used in an earlier implementation of the SCI Gateway. This attribute is now obsolete and ignored.

4.2.4.3.5 "check"

Questions defined as type='check', will be converted to an HTML checkbox (input type="checkbox") element on the message screen.

Attribute	Usage	Description
filters	Optional (lower case)	The filter node specifies what information will be retrieved from the Clinical events collection of the patient.
help	Optional (lower case)	The help text.
hidden	Optional (lower case)	Hides the control's label if set to true.
hint	Obsolete (lower case)	Used in an earlier implementation of SCI Gateway. These nodes are now obsolete and ignored.
id	Mandatory (lower case)	Name of the HTML checkbox.
mandatory	Optional (lower case)	Indicates that the question is mandatory and the user cannot submit the protocol until it is completed. An asterix (*) will be added after the label to indicate the mandatory nature.
object	Optional (lower case)	Generic pre-population object.
objectOld	Optional (camel case)	GPASS pre-population object.
property	Optional (lower case)	Generic pre-population object's property.
propertyOld	Optional (camel case)	GPASS pre-population object's property.
readonly	Optional (lower case)	Indicates that the checkbox is read only and cannot be edited directly by the user. The read only state for a question is only activated if the question has been pre-populated by a 3rd party application (i.e. GPASS). It is most applicable for patient identification data such as CHI.
text	Mandatory (lower case)	Label of the HTML checkbox on the web-form.
userName	Optional (camel case)	A friendly name for the question. This is only ever used in conjunction with a mandatory="true" attribute and is used in the error message to inform the user that the mandatory field has not been filled in.
validate_type	Obsolete (lower case)	Used in an earlier implementation of the SCI Gateway. This attribute is now obsolete and ignored.

4.2.4.3.6 "radio"

Questions defined as type='radio', with options as child nodes, will be converted to an HTML radio button group on the message screen.

Attribute	Usage	Description
filters	Optional (lower case)	The filter node specifies what information will be retrieved from the Clinical events collection of the patient.
help	Optional (lower case)	The help text.
hidden	Optional (lower case)	Hides the control's label if set to true.
hint	Obsolete (lower case)	Used in an earlier implementation of SCI Gateway. These nodes are now obsolete and ignored.
id	Mandatory (lower case)	Name of the HTML radio button group. Note: Each radio button in the group will have this as its HTML name attribute. The HTML ID attribute will be of the following format: [name]_[sequence_number]
mandatory	Optional (lower case)	Indicates that the question is mandatory and the user cannot submit the protocol until it is completed. An asterix (*) will be added after the label to indicate the mandatory nature.
object	Optional (lower case)	Generic pre-population object.
objectOld	Optional (camel case)	GPASS pre-population object.
option	Optional (lower case)	Option nodes are the entries in the radio group. At least one option node should be specified to allow the selection of at least one option.
property	Optional (lower case)	Generic pre-population object's property.
propertyOld	Optional (camel case)	GPASS pre-population object's property.
text	Mandatory (lower case)	Label of the HTML radio button group on the web-form.
userName	Optional (camel case)	A friendly name for the question. This is only ever used in conjunction with a mandatory="true" attribute and is used in the error message to inform the user that the mandatory field has not been filled in.

4.2.4.3.7 "table"

Questions defined as type='table', with columns and rows as child nodes, will be converted to an HTML table with static columns and rows on the message screen.

Attribute	Usage	Description
column	Optional (lower case)	At least one column should be specified.
help	Optional (lower case)	The help text.
id	Mandatory (lower case)	Name of the HTML table.
object	Optional (lower case)	Pre-population API object.
objectOld	Optional (camel case)	Pre-population GPASS object.
row	Optional (lower case)	At least one row should be specified.
text	Mandatory (lower case)	Label of the HTML table on the web-form.

4.2.4.3.8 "dynaTable"

Questions defined as type='dynaTable', with columns as child nodes, will be converted to an HTML table with static columns and dynamic rows on the message screen.

Attribute	Usage	Description
column	Optional (lower case)	At least one column should be specified.
filters	Optional (lower case)	The filter node specifies what information will be retrieved from the Clinical events collection of the patient.
help	Optional (lower case)	The help text.
id	Mandatory (lower case)	Name of the HTML table.
object	Optional (lower case)	Pre-population API object.
objectOld	Optional (camel case)	Pre-population GPASS object.
text	Mandatory (lower case)	Label of the HTML table on the web-form.

4.2.4.3.9 "hidden"

Attribute	Usage	Description
id	Mandatory (lower case)	Name of the hidden HTML text field.
object	Optional (lower case)	Generic pre-population object.
objectOld	Optional (camel case)	GPASS pre-population object.
property	Optional (lower case)	Generic pre-population object's property.
propertyOld	Optional (camel case)	GPASS pre-population object's property.
readonly	Optional (lower case)	Indicates that the hidden text field is read only and cannot be edited directly by the user. The read only state for a question is only activated if the question has been pre-populated by a 3rd party application (i.e. GPASS). It is most applicable for patient identification data such as CHI.
text	Mandatory (lower case)	Label of the HTML hidden HTML text field on the web-form. This label will be hidden.

4.2.4.4. OBJECTS USED BY VARIOUS QUESTION TYPES

Various objects are used in the definition of the different question types. These objects are:

- option
- filters (these contain filtergroup elements which contain filter elements)
- row
- column

4.2.4.4.1 Option

Option elements are used for the specification of options of "select" and "radio" question elements. These option elements specify the options that the user can choose from.

Attribute	Usage	Description
filters	Optional (lower case)	The filter node specifies what information will be

		retrieved from the Clinical events collection of the patient.
hidden	Optional (lower case)	This hides the specified option.
id	Mandatory (lower case)	Name of the option.
isDefault	Optional (camel case)	When this property is set to “true”, the option is selected as the default option.
source	Optional (lower case)	This property corresponds with values in the primary care system, which allows the option to be selected.
text	Mandatory (lower case)	The text that is displayed for this option in the dropdown list.
value	Optional (lower case)	The value that is used if this option is chosen to be submitted.

4.2.4.4.2 Filter

Filter elements are used for the specification of criteria for pre-population from the patient’s clinical events collection.

Attribute	Usage	Description
comparison	Optional (lower case)	Type of comparison that needs to be performed on the value of the specified property (equals, lessthan, atleast, morethan, atmost). If not specified the default is equals.
property	Mandatory (lower case)	A property of the ReadCodeEvent, ScreeningEvent or PrescribedDrug object.
type	Optional (lower case)	
value	Mandatory (lower case)	The value that needs to be compared against.

4.2.4.4.3 Row

Row elements are used for the specification of the rows of question elements of the **table** question type.

Attribute	Usage	Description
filter	Optional (lower case)	Filter that specifies what information from the patient’s clinical events collection is retrieved.
text	Mandatory (lower case)	Label for the row within the HTML table.

4.2.4.4.4 Column

Column elements are used for the specification of the columns of question elements of the **dynaTable** and **table** question type.

Attribute	Usage	Description
id	Mandatory (lower case)	The unique identifier of the column.
mandatory	Optional (lower case)	The mandatory nature of the column.
property	Optional (lower case)	Generic pre-population object’s property.
propertyOld	Optional (camel case)	GPASS pre-population object’s property.
spellCheck	Optional (camel case)	Can the question be spell checked. If the question is also readonly then spell checking will be disabled.
text	Mandatory (lower case)	The header of the column.

type	Optional (lower case)	This attribute only needs to be set for date columns.
userName	Optional (camel case)	The user friendly message that will be displayed if the field is mandatory and empty.

4.2.4.5. QUESTION TYPE – USING “SELECT” AND “RADIO” TYPES

Drop down lists and radio groups are often very effective within protocols when the data required can be pre-defined within a set of options. Typically when the number of options is small (e.g. two or three) then radio groups are most effective. For larger selections, drop down lists are better suited. Both are implemented in a similar way within SCI protocols.

To demonstrate their use, here are two variations of a question to ask for a colour to be entered. The first example uses the “radio” input type – just an extract is shown here the rest of the protocol definition has been omitted.

```
…
<question text="Colour" type="radio" id="colour">
        <option text="(Not Known)" value="Not Known" id="colour_Not_Known"
isDefault=”true”/>
        <option text="Red" value="Red" id="colour_Red"/>
        <option text="Green" value="Green" id="colour_Green"/>
        <option text="Blue" value="Blue" id="colour_Blue"/>
        <option text="Other" value="Other" id="colour_Other"/>
</question>
…
```

When a choice has been selected and the protocol form submitted, the text value held within the attribute “value” is passed through as the data. For example, if “(Not Known)” was selected, the value passed would be “Not Known”. This allows the displayed choice to be different from the processed value – useful if using codified values such as health board ciphers but the user would need to see the non-codified value. The following shows how the “radio” type appears within the protocol web-form:

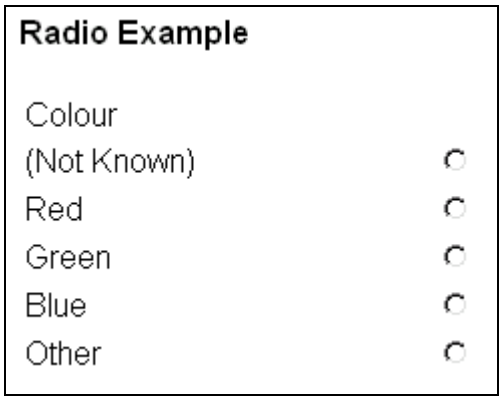


Figure 5 - Using the "radio" question type

The second example asks the same question using the “select” question type – just an extract is shown here the rest of the protocol definition has been omitted.

```
…
<question text="Colour" type="select" id="colour">
        <option text="(Not Known)" value="Not Known" id="colour_Not_Known"/>
        <option text="Red" value="Red" id="colour_Red"/>
        <option text="Green" value="Green" id="colour_Green"/>
        <option text="Blue" value="Blue" id="colour_Blue"/>
```

```
        <option text="Other" value="Other" id="colour_Other"/>
</question>
…
```

Such questions are processed in the same way as with “radio” groups with the data held in the “value” attribute processed by the SCI Gateway as the selected value. The following shows how the “select” type appears within the protocol web-form:

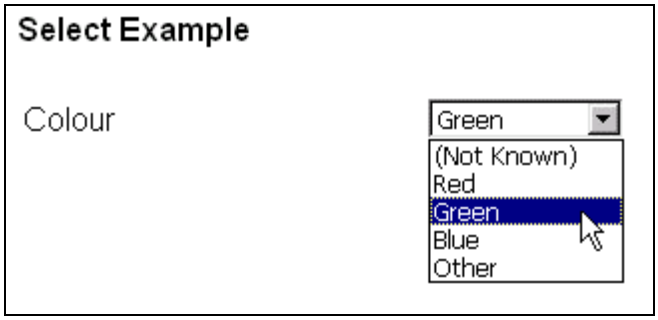


Figure 6 - Using the "select" question type

4.2.4.6. QUESTION TYPE – USING THE “CHECK” TYPE

Check boxes allow for quick and simple yes/no responses to questions. They provide an effective means for capturing clinical questions and reduce the need for manual data input through the keyboard.

```
…
<category name="diagnosis" text="Diagnosis, check all that apply.">
        <question text="Leg pain" type="check" id="examination_leg_pain"/>
        <question text="Radiating To foot" type="check"
id="examination_radiating_to_foot"/>
        <question text="Altered reflexes" type="check"
id="examination_altered_reflexes"/>
        <question text="Simple low back pain" type="check"
id="examination_simple_back_pain"/>
</category>
…
```

How the “check” type appears within the protocol web-form is shown below. Such questions often work best when contained within a sub-category. This allows the title of the category to act as a heading for the selection of questions to follow.

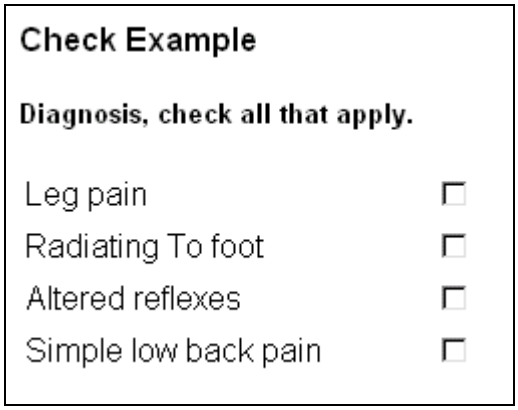


Figure 7 - Using the "check" question type

When submitted through the SCI Gateway, the text used for the check box is passed through as the data. For example, if the “Leg pain” check box was selected the data passed would be “Leg pain” and this text would appear on the referral letter and/or passed to the hospital Outpatients system.

4.2.4.7. QUESTION TYPE – USING “TABLE” AND “DYNATABLE” TYPES

Tables are the most complex type of question supported by a SCI protocol. Two types of table are possible, a static table and a dynamic table.

Recommendation: The use of tables is closely linked to the way the SCI Gateway handles protocols, therefore protocol authors are best advised to reuse table definitions found in existing protocols, rather than create new table definitions from scratch.

The example below contains both types of table definition.

```
<category name="tableExamples" text="Table Examples">
        <category name="currentMedication" text="Current Medication (DynaTable)">
                <question text="Administered Medication" type="dynaTable"
id="administeredMedication">
                        <column name="DrugName" text="Drug Name" id="DrugName"/>
                        <column name="preparation" text="Preparation" id="preparation"/>
                        <column name="dose" text="Dose" id="dose"/>
                        <column name="startDate" text="Start Date" id="startDate"
type=”date”/>
                        <column name="frequency" text="Frequency" id="frequency"/>
                        <column name="quantity" text="Quantity" id="quantity"/>
                </question>
        </category>
        <category name="diagnosis" text="Diagnosis (Table)">
                <question text="Administration Alerts" type="table" id="admin_alert">
                        <row text="Visual Impairment"><filter property="readCode"
comparison="pattern" value="F49.."/></row>
                <row text="Impaired Hearing"><filter property="readCode"
comparison="pattern" value="F59.."/></row>
                        <column text="Description" id="comment"/>
                        <column text="Recorded Date" id="date" type=”date”/>
                </question>
        </category>
</category>
```

The additional XML elements used for **table** definitions are `<column>` and `<row>`. The additional XML elements used for **dynatable** definitions are `<column>`. Both these elements have similar constructs with a hidden system name (“id”) and a display name (“text”).

The <filter> child element of the `<row>` element is used to extract information from the GPASS system. This functionality is covered in the section ‘Auto-Population of Protocol Questions from the GPASS System’.

How the tables appear within the protocol web-form is shown below. A dynamic table is shown with an “Add Row” button that allows the user to manually add an extra row to the table for more data entry.

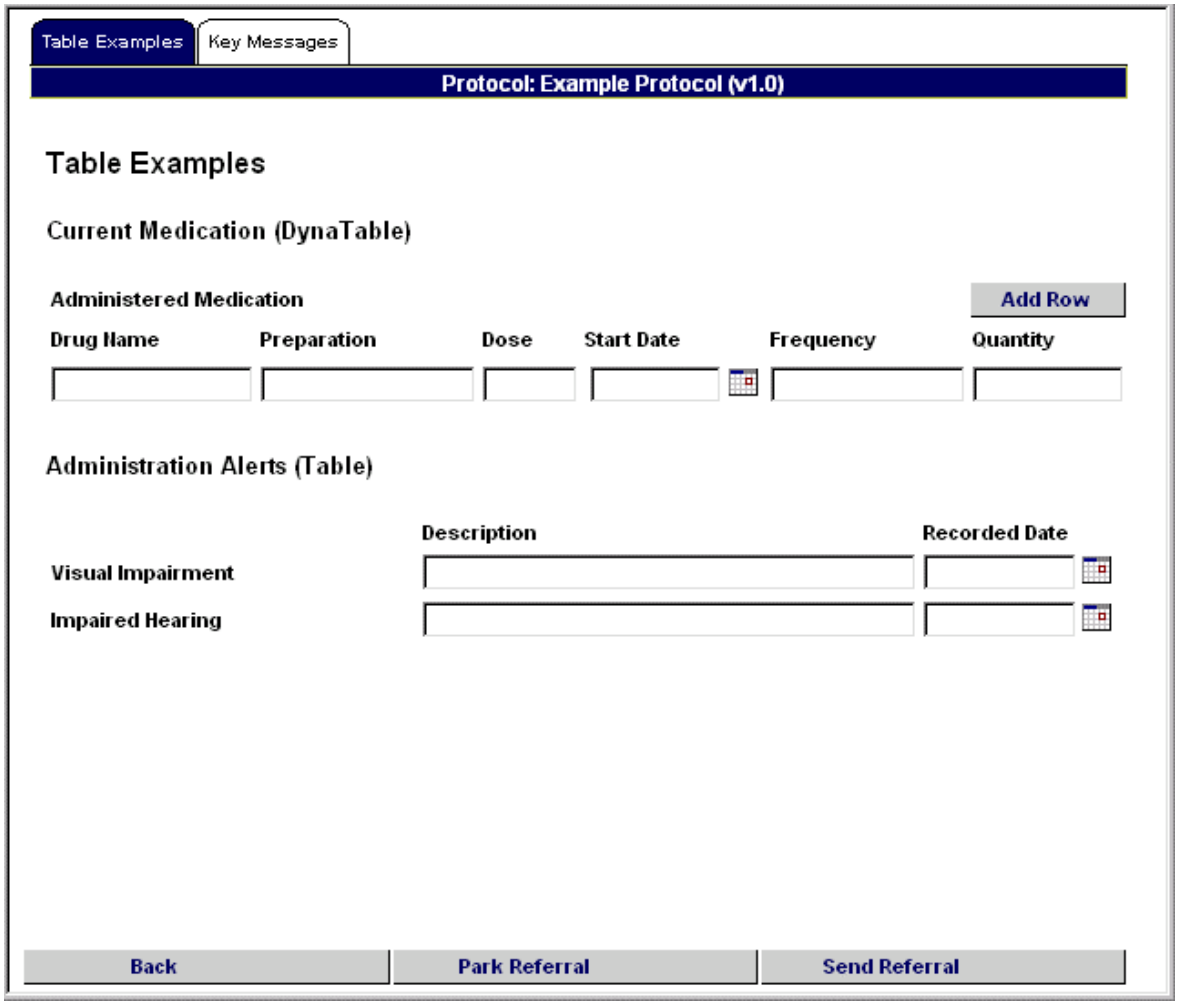


Figure 8 - Using "table" and "dynaTable" question types

The type attribute of the column set to date, indicates that the column is used for a date value and a date picker control is shown beside the column. Eg. <column text="Recorded Date" id="date" type=”date”/>

4.2.5. Element “help”

Pop-up help messages can be incorporated into the protocol by using the `<help>` element inside a `<question>` element. The `<help>` element defines the text to display in a pop-up message box activated when the user clicks on a “Help” link on the web-form.

The example below shows how to use the `<help>` element. A special syntax is used to add a carriage return into the message that can improve the layout and readability of help messages. The syntax to add a carriage return is “#NEWLINE_”.

```
…
<question text="Example Question" type="text" id="test">
        <help>
        Help text can be displayed within pop-up message #NEWLINE_boxes for any question defined within the protocol.
        </help>
</question>
…
```

How pop-up help messages are displayed to the user is shown below.

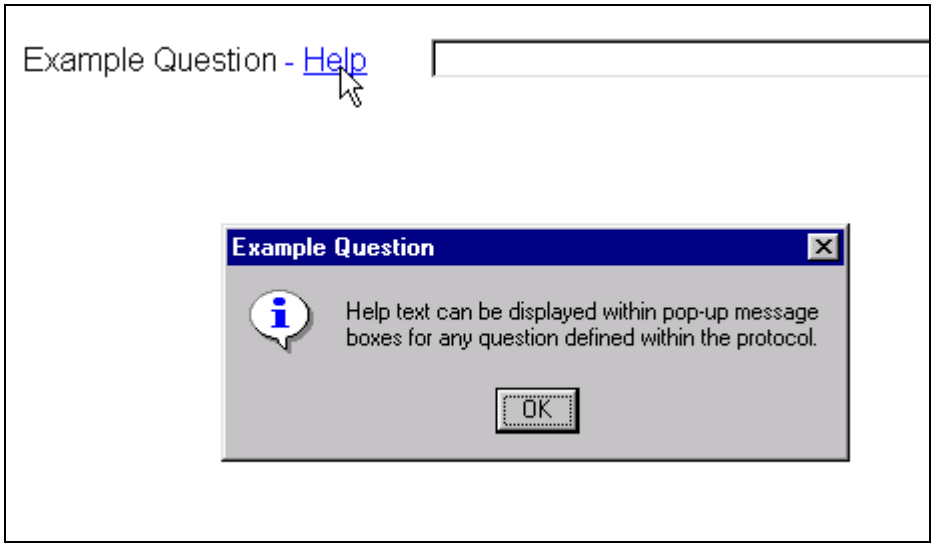


Figure 9 - Add Help Text to protocols

4.2.6. Element "freeText"

Another mechanism to include supporting or informative content into a protocol is to use the `<freeText>` element. Within this element, the author can write standard HTML script thus allowing more control and formatting possibilities.

Such information is most effective when contained within it's own top-level category, thus appearing within a dedicated tab. The use of the tab heading "Key Messages" is becoming a standard tab for all protocols that include supporting or informative content.

Scripting HTML is a common development skill however there are a few key points to remember when using HTML within the `<freeText>` element for SCI protocols:

- The HTML tags `<html>` and `<body>` do not need to be included
- No external files can be referenced therefore you cannot use `<img>` or `<link>` tags
- All HTML tags must be properly closed. Unlike XML you can normally leave HTML tags unclosed without causing errors. This is bad coding practice but most implementations of HTML allow this to happen. Within the SCI environment, as the HTML is embedded within XML, all tags must be closed. A common HTML tag written incorrectly will be `<br>` that adds a link break. This should be written as either `<br/>` or `<br></br>`
- Nesting of HTML tags must be correct. In HTML, most implementations will allow incorrectly nested elements (e.g. some text). Within SCI environment, the tags must be nested correctly (e.g. some text).

An example of using the `<freeText>` element is provided below.

```
<category name="keyMessages" text="Key Messages">
        <freeText>
                <table border="1" width="90%">
                        <tr>
                                <td>
                                        <font size="4" color="red">Red Flags For
<u>Emergency</u></font>
                                        <br/>
```

```
                                        <font size="3" color="blue">Orthopaedic
Referral</font>
                                </td>
                        </tr>
                        <tr>
                                <td>
                                        <li>Bladder/bowel disturbance</li>
                                        <li>Saddle anasthaesia</li>
                                        <li>Gait disturbance</li>
                                </td>
                        </tr>
                </table>
                <br/>
                <table border="1" width="90%">
                        <tr>
                                <td>
                                        <font size="4" color="red">Red Flags For
<u>Urgent</u></font>
                                        <br/>
                                        <font size="3" color="blue">Referral/Combined Back
Clinic Referral</font>
                                </td>
                        </tr>
                        <tr>
                                <td>
                                        <li>Systemically unwell</li>
                                        <li>Previous cancer diagnosis</li>
                                        <li>Possible infectious cause</li>
                                        <li>New/altered back pain</li>
                                        <li>Age less than 20yrs or over 50yrs</li>
                                        <li>Raised ESR/abnormal</li>
                                        <li>FBC/LFTs</li>
                                </td>
                        </tr>
                </table>
        </freeText>
</category>
```

How this appears within the protocol web-form is shown below.

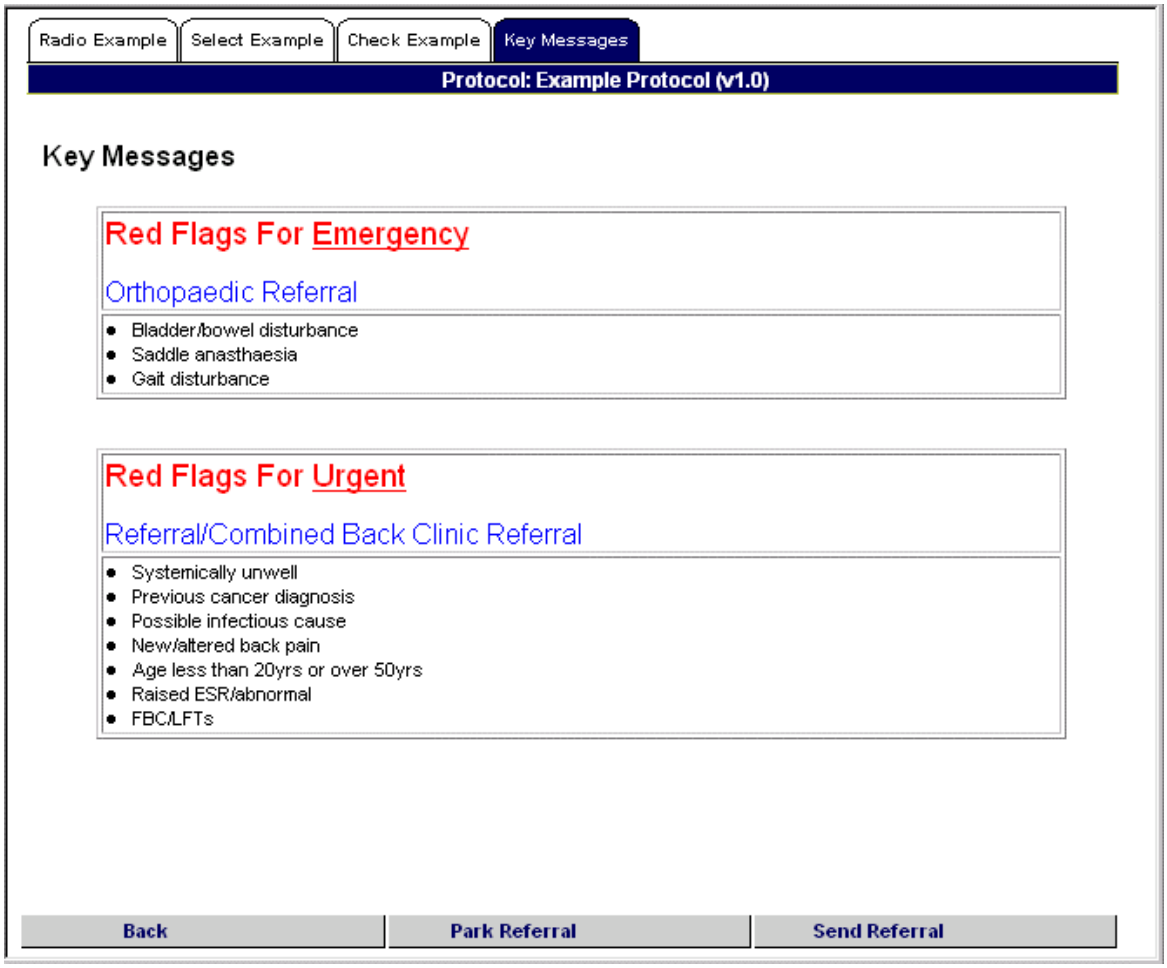


Figure 10 - Adding supportive or informative content to protocols

All content within the `<freeText>` element is ignored when the protocol is processed by the SCI Gateway application. Therefore the use of the `<freeText>` element should be for static information only and not for any data entry.

4.3. Mandatory Fields

Within SCI Gateway there are two levels of mandatory fields. The first level is based on the national schemas. The second level is based on requirements by the SCI Gateway application.

4.3.1. Referral Mandatory Fields at Schema Level

From R10.2, the SCI Gateway uses the national message schemas to validate any message. This section will describe the mandatory fields required in all referral screen definitions.

The national referral schema for Scotland defines the following **fields** to be mandatory:

Field Identifiers
maincondition_readcode or maincondition_description
referral_date
Referraltype

referral_expectedoutcome
Priority

In addition to these fields being mandatory, the labels of dynamic fields (fields whose identifiers are prefixed with either **examination_** or **investigation_**) are mandatory as well.

The national referral schema for Scotland defines the following **columns** in **tables** to be mandatory:

Tables	
Table Identifier	**Column Identifier**
administeredmedication, pastmedication	Drugname
Pastcondition	Conditionname
Pastprocedure	Conditionname
Familycondition	Conditionname
Examination_	Description
Investigation_	Description

4.3.2. Mandatory Fields at Application Level

At application level, the SCI Gateway has a number of mandatory fields as well for patient identification for both primary and secondary care. These are the following:

Field Identifiers	Notes
Chi	
patient_forename	
patient_surname	
patient_sex	This question must be of type radio, select or hidden. Values must be 'M' or 'F' ('male' & 'female' are converted to 'M' & 'F' respectively). Any other values and the patient_sex is stored as an empty string.
patient_dob	

4.4. Referrer

The individual referring a patient using the SCI Gateway can be located at a primary or secondary care from R10.2. This means that the referrer can be a general practitioner or a hospital consultant.

In order to facilitate this requirement the following identifiers have been implemented:

Primary Care	Description
refGp_gmc_code	Referring gp gmc code
refGp_practice_Code	Referring gp practice code
refGp_code	Referring gp gpcode
refGp_name	Referring gp name
refGp_practiceName	Referring gp practice name
refGp_practice_addressLine	Referring gp practice address
refGp_practice_phone	Referring gp practice phone

refGp_practice_fax	Referring gp practice fax
refGp_practice_email	Referring gp practice email

Secondary Care	Description
refhcp_scheme	Referring health care professional scheme
refhcp_gmc_code	Referring health care professional gmc code
refhcp_forename	Referring health care professional forename
refhcp_surname	Referring health care professional surname
reforg_description	Referring health care organisation description
reforg_location_code	Referring health care organisation location code
reforg_name	Referring health care organisation name
reforg_address	Referring health care organisation address
reforg_postcode	Referring health care organisation postcode

4.5. Referral GP/HCP Dropdown

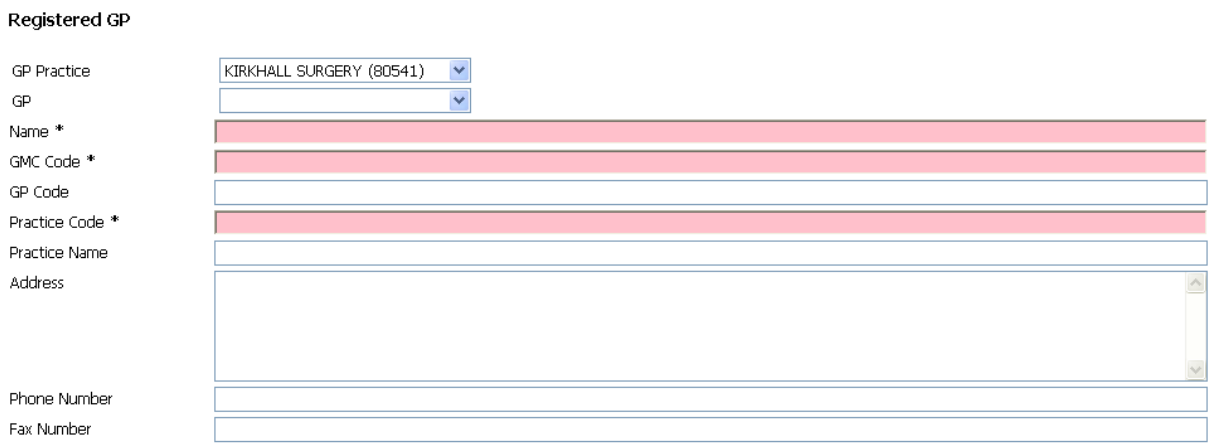
4.5.1. Referral GP Dropdown

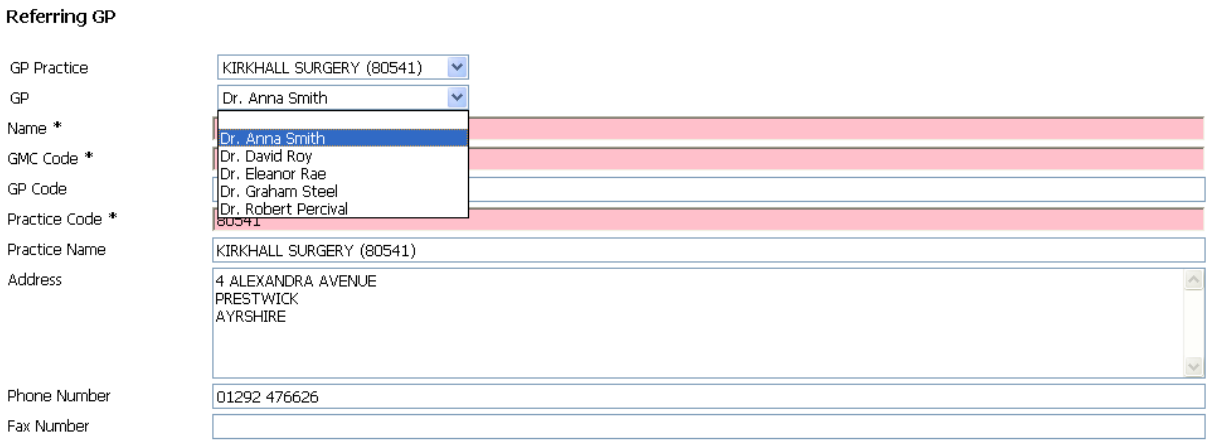
The Administration tab can be configured to display dropdown lists for the GP Practice and GP within the Registered GP and Referring GP sections. To enable this functionality the control element should be configured within the protocol xml:

```
<category text="Registered GP">
        <control name="registrar"/>
        <question text="Name" type="text" id="reggp_name" mandatory="true" userName="Reg GP Name - Pat Admin Tab"
object="registeredclinician" property="Name"/>
        …
</category>

<category text="Referring GP">
        <control name="referrer"/>
        <question text="Name" type="text" id="refgp_name" mandatory="true" userName="Ref GP Name - Pat Admin Tab"
object="referringclinician" property="Name"/>
        …
</category>
```

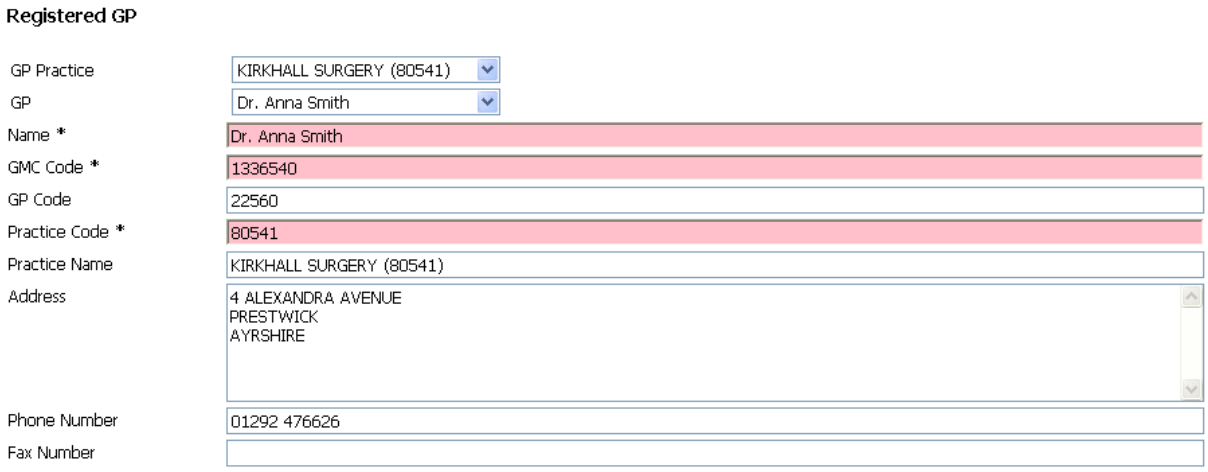
For users configured at GP practice level the following details will be shown:

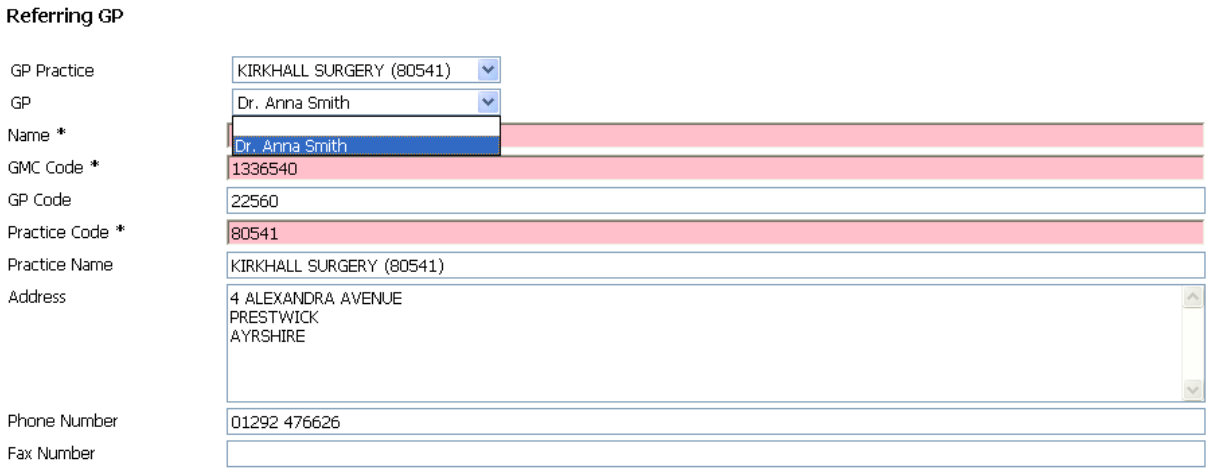




The GP practice dropdown will be pre-populated and the user will be able to select the required GP from the GP dropdown. This will then populate the additional fields.

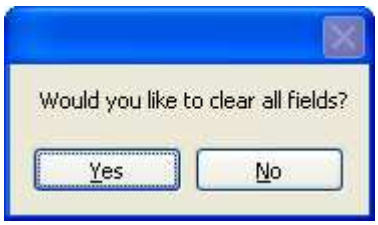
For users configured at GP level the following details will be shown:





The GP practice, GP and additional fields will be pre-populated.

To clear the additional fields the blank entry within the GP dropdown should be selected and the following message will be displayed:

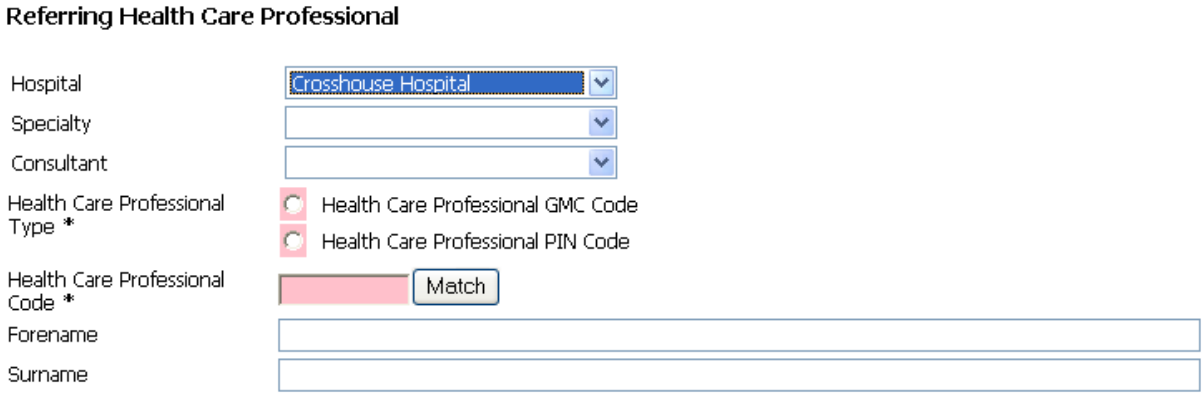


4.5.2. Referral HCP Dropdown

The Administration tab can be configured to display dropdown lists for the Hospital, Specialty and Consultant within the Referring Health Care Professional section. To enable this functionality the control element should be configured within the protocol xml:

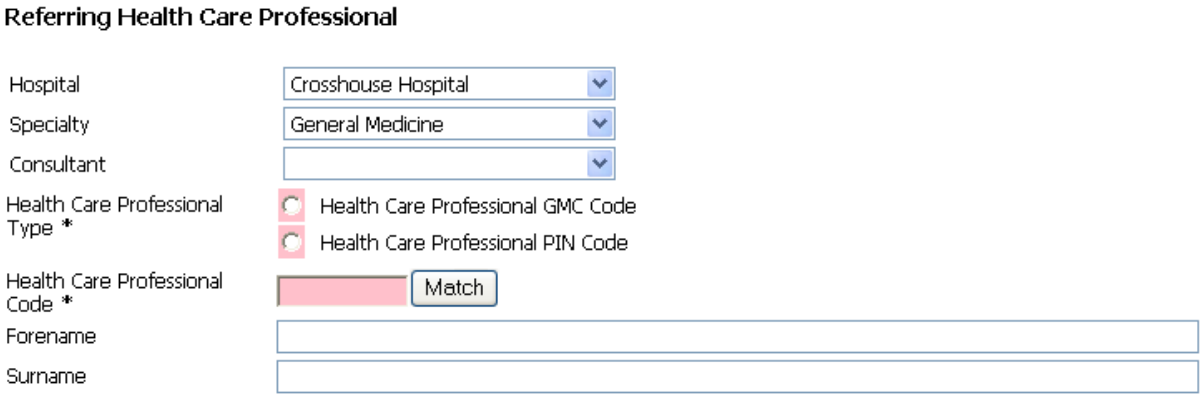
```
<category text="Referring Health Care Professional">
        <control name="referrer"/>
        <question text="Health Care Professional Type" type="radio" id="refhcp_scheme" mandatory="true" userName="HCP Type" onclick="HcpSchemeChanged()">
                <option text="Health Care Professional GMC Code" value="GmcCode" source="1"/>
                <option text="Health Care Professional PIN Code" value="PinCode" source="2"/>
        </question>
        …
</category>
```

For users configured at Hospital level the following details will be shown:



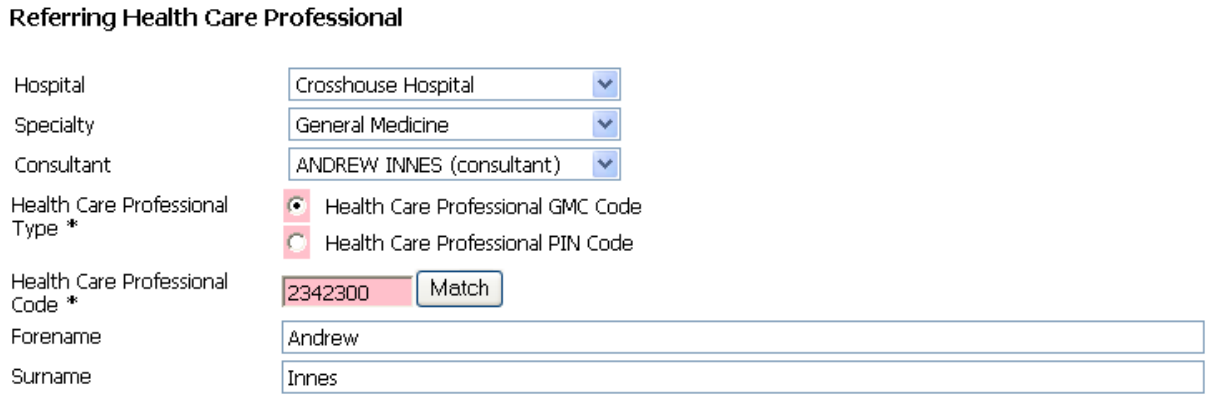
The Hospital dropdown will be pre-populated and the user will be able to select the required specialty and consultant. This will then populate the additional fields including the Referring Organisation details.

For users configured at Specialty level the following details will be shown:



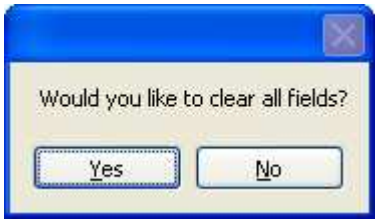
The Hospital and Specialty dropdowns will be pre-populated and the user will be able to select the required consultant. This will then populate the additional fields including the Referring Organisation details.

For users configured at Consultant level the following details will be shown:



The Hospital, Specialty, Consultant and the additional fields, including the Referring Organisation details, will be pre-populated.

To clear the additional fields the blank entry within the Specialty or Consultant dropdowns should be selected and the following message will be displayed:



4.6. Auto-Population of Protocol Questions from the GPASS System

The SCI Gateway and the implementation of SCI protocols is integrated with the GPASS system used extensively within primary care. As the vast quantity of protocols will be written to support electronic referrals and many of these will be sent by GP's, the system has been written to auto-populate many of the patient demographic and medical data fields common to most referral protocols.

4.6.1. Using the "objectOld" and "propertyOld" Attributes

If the field being described in a `<question>` element is to be automatically populated from GPASS then the "objectOld" and "propertyOld" elements must be provided. An example of using such a construct is provided below. The example shows extracting the patient's surname from GPASS.

```
<question text="Surname" type="text" id="patient_surname"  objectOld="patient"
propertyOld="surname">
</question>
```

This is how the system knows where to look for the data within the GPASS system.

Note: Pre-population of the maincondition_onsetdate can only be set up by specifying objectOld=”Referral” and propertyOld=”StartDate”. The onset date is not supported within GPASS.

Refer to Appendix A – GPASS Enumerations - for a list of the currently supported extraction fields from GPASS.

4.6.2. Using the “filter" Element for Lists and Radio Groups

Another mechanism for extracting information from the GPASS system is to add a <filters type=”GPASS”> element, and a <filter> element to `<option>` elements used for “select” and “radio” question types.

The value of the “value” attribute of the <filter> element would be a valid clinical read code as used by GPASS. For example `value=”1376.”` denotes a very heavy smoker. If the patient record contains the defined read code then the associated question value is automatically selected.

```
<question text="Smoking" type="select" id="riskSmoking" object="ReadCodeEvent">
        <option text="(Not Known)" value="Not Known"/>
        <option text="Never Smoked Tobacco" value="Never smoked tobacco">
                <filters type="GPASS">
                        <filter property="readCode" comparison="pattern" value="1371."/>
                </filters>
        </option>
        <option text="Trivial Smoker (&lt;1 cig/day)" value="Trivial smoker">
                <filters type="GPASS">
                        <filter property="readCode" comparison="pattern" value="1372."/>
                </filters>
        </option>
        <option text="Light Smoker (1-9 cigs/day)" value="Light smoker">
                <filters type="GPASS">
                        <filter property="readCode" comparison="pattern" value="1373."/>
                </filters>
        </option>
        <option text="Moderate Smoker (10-19 cigs/day)" value="Moderate smoker">
                <filters type="GPASS">
                        <filter property="readCode" comparison="pattern" value="1374."/>
                </filters>
        </option>
        <option text="Heavy Smoker (20-39 cigs/day)" value="Heavy smoker">
                <filters type="GPASS">
                        <filter property="readCode" comparison="pattern" value="1375."/>
                </filters>
        </option>
        <option text="Very Heavy Smoker (40+ cigs/day)" value="Very heavy smoker">
                <filters type="GPASS">
                        <filter property="readCode" comparison="pattern" value="1376."/>
                </filter>
        </option>
</question>
```

This example provided shows a simplified version of this question as used by existing SCI protocols. Within the full question, extra options for ex-smoking classifications are included.

If the comparison attribute is set (i.e. has a value of “pattern”) the value entered by the user will be evaluated to ensure it is correct. The value of the “value” attribute shows the Regular Expression (or RegEx) used to validate the data. This functionality is covered in the Regular Expressions section.

4.6.3. Read Codes in Filter Elements

Filter elements allow pre-population from the clinical events collection that is available for a particular patient. The clinical events collection consists of ReadCodeEvents, ScreeningEvents, and PrescribedDrugs.

Valid values for the filter element’s **property** attribute are the properties of the ReadCodeEvents, ScreeningEvents, and PrescribedDrug objects.

Valid values for the filter element’s **comparison** attribute are:

equals
lessthan
atleast
morethan
atmost
days
pattern
notpattern

Valid values for the filter element’s **value** attribute are:
[readcodes]
[regular expressions]
[number of records to return]

The table below shows the most commonly used read codes for pre-population from GPASS.

Description	Readcodes	Priority	Date Started	End Date	Is Active
Tobacco consumption	137.. (1371.- 137Z.)	x		X	
Alcohol consumption	136.. (1361. – 136Q.)	x		X	
Exercise consumption	138..(1381. – 1386.)	X		X	
Past conditions	[A-Q]....	< 1		0	
Past procedures	[3-8]....	< 2		< 365	
Family history	(ZV1([6-9]|A).)|12...	< 1			
Administered medication					True
Past medication			90 days		False

Description	Readcodes	Priority	Date Started	End Date	Is Active
Allergy alert	ZV(14.|150|u6[MNPT]) |14[LM]..|J432.|SN58.				
Intolerances alert	C31[23].|J69y[45z]|U60..				

The example below shows how the pre-existing conditions table is set up for auto-population from GPASS.

```
<question text="Pre-existing Conditions" type="dynaTable" id="pastcondition" objectOld="ReadCodeEvent">
    <column text="Condition Name" id="conditionname" mandatory="true" propertyOld="ReadCodeDescription"/>
    <column text="Laterality" id="laterality" property="laterality"/>
    <column text="Modifier" id="modifier" propertyOld="ReadCodeModifier" property="modifier"/>
    <column text="Extension" id="comment" propertyOld="ReadCodeMeaningExtension" property="comment"/>
    <column type="Date" text="Date of Onset" id="date" property="date"/>
    <filters type="GPASS">
        <filter property="ReadCode" comparison="pattern" value="[A-Q]...."/>
        <filter property="Priority" comparison="lessthan" value="1"/>
        <filter property="EndDate" comparison="days" value="0"/>
    </filters>
</question>
```

4.7. Auto-Population of Protocol Question from other GP systems

The SCI Gateway and the implementation of SCI protocols is integrated with any General Practitioner system that has implemented the SCI Gateway GP Third Party Programming Interface.

As the vast quantity of protocols will be written to support electronic referrals and many of these will be sent by GP's, the system has been written to auto-populate many of the patient demographic and medical data fields common to most referral protocols.

4.7.1. Using the "object" and "property" Attributes

If the field being described in a `<question>` element is to be automatically populated from GP systems that use the generic auto-population, then the "object" and "property" elements must be provided. An example of using such a construct is provided below. The example shows extracting the patient's surname from a GP system other than GPASS.

```
<question text="Surname" type="text" id="patient_surname"  object="patient"
property="surname">
</question>
```

This is how the system knows where to look for the data within the GP system.

Refer to Appendix B – Other GP Systems Enumerations - for a list of the currently supported extraction fields from GP systems that use the generic auto-population.

4.7.2. Using the "filter" Element for Lists and Radio Groups

Another mechanism for extracting information from other GP systems is to add a <filters type="Generic"> element, a filtergroup child element with the coding scheme specified in the scheme attribute, and a <filter> element to `<option>` elements used for "select" and "radio" question types.

The value of the "value" attribute of the <filter> element would be a valid clinical code for the specified coding scheme in the filtergroup (scheme attribute) as used by third party application. For example if the scheme attribute specified is "Read" for the read clinical

coding scheme, the value attribute could be `value="1376."` This denotes a very heavy smoker. If the patient record contains the defined clinical code then the associated question value is automatically selected.

```
<question text="Smoking" type="select" id="riskSmoking" object="ReadCodeEvent">
        <option text="(Not Known)" value="Not Known"/>
        <option text="Never Smoked Tobacco" value="Never smoked tobacco">
                <filters type="Generic">
                        <filtergroup scheme="Read">
                                <filter property="readCode" comparison="pattern" value="1371."/>
                        </filtergroup>
                </filters>
        </option>
        <option text="Trivial Smoker (&lt;1 cig/day)" value="Trivial smoker">
                <filters type="Generic">
                        <filtergroup scheme="Read">
                                <filter property="readCode" comparison="pattern" value="1372."/>
                        </filtergroup>
                </filters>
        </option>
        <option text="Light Smoker (1-9 cigs/day)" value="Light smoker">
                <filters type="Generic">
                        <filtergroup scheme="Read">
                                <filter property="readCode" comparison="pattern" value="1373."/>
                        </filtergroup>
                </filters>
        </option>
        <option text="Moderate Smoker (10-19 cigs/day)" value="Moderate smoker">
                <filters type="Generic">
                        <filtergroup scheme="Read">
                                <filter property="readCode" comparison="pattern" value="1374."/>
                        </filtergroup>
                </filters>
        </option>
        <option text="Heavy Smoker (20-39 cigs/day)" value="Heavy smoker">
                <filters type="Generic">
                        <filtergroup scheme="Read">
                                <filter property="readCode" comparison="pattern" value="1375."/>
                        </filtergroup>
                </filters>
        </option>
        <option text="Very Heavy Smoker (40+ cigs/day)" value="Very heavy smoker">
                <filters type="Generic">
                        <filtergroup scheme="Read">
                                <filter property="readCode" comparison="pattern" value="1376."/>
                        </filtergroup>
                </filters>
        </option>
</question>
```

This example provided shows a simplified version of this question as used by existing SCI protocols. Within the full question, extra options for ex-smoking classifications are included.

If the comparison attribute is set (i.e. has a value of “pattern”) the value entered by the user will be evaluated to ensure it is correct. The value of the “value” attribute shows the Regular

Expression (or RegEx) used to validate the data. This functionality is covered in the Regular Expressions section.

4.7.3. Auto-population for Tables

The SCI Gateway third party interface xml schema determines the patient details for generic auto-population.

In this schema, the clinical information about the patient is categorised into the following sections:

1. Medication
   1.1. Current medication
   1.2. Past medication
2. Conditions
3. Procedures
4. FamilyConditions
5. Alerts
   5.1 Allergies
   5.2 Intolerances
   5.3 Risks to others

As the SCI Gateway protocols have controls for these sections, a mapping between the patient details xml and the protocol xml exists:

Table identifier	Xml section	Object attribute value
administeredmedication	Current medication	prescribeddrug
pastmedication	Past medication	pastmedication
pastcondition	Conditions	Conditions
pastprocedure	Procedures	Procedures
familycondition	FamilyConditions	familycondition
alert_allergy	Allergies	Allergies
alert_intolerances	Intolerances	Intolerances
alert_risktoothers	Risks to others	RiskToOthers

The auto-population for these tables is set up using the object and property attributes for the question. Below is an example:

```
<question text="Pre-existing Conditions" type="dynaTable" id="pastcondition" object="Conditions">
    <column text="Condition Name" id="conditionname" mandatory="true" property="conditionname"/>
    <column text="Laterality" id="laterality" property="laterality"/>
    <column text="Modifier" id="modifier" property="modifier"/>
    <column text="Extension" id="comment" property="comment"/>
    <column type="Date" text="Date of Onset" id="date" property="date"/>
</question>
```

Refer to Appendix B – Other GP Systems Enumerations - for a list of the currently supported tables and their columns from GP systems that use the generic auto-population.

Once the **object** and **property** attributes have been set up, the clinical information for that table will be extracted from the GP system that uses the generic auto-population.

4.7.4. Restricting Auto-population for Tables

When a table has been set up for auto-population, all the information from the relevant section will be auto-populated. This means that once the past conditions table has been set up with object and property attributes, **all** the patient's conditions will auto-populate.

As certain referrals will only require specific information to come through, the filter mechanism has been introduced.

Filters

The first step for defining a filter for generic auto-population will be to create a **filters** child element of the question. This **filters** element is required to have a **type** attribute with a value of '**Generic**'.

Optionally, the **filters** element can also have a **count** attribute. This restricts the auto-population to only bring through the most recent x amount of rows (based on the recorded date).

Within the **filters** element, any number of **filtergroup** elements can be defined. It might be necessary to do so because of the different coding schemes and systems being used in GP practices. Filtergroups are OR'ed.

Filter groups

The **scheme** attribute is mandatory, and it indicates what coding scheme is used within the filtergroup. The **system** attribute is optional, and is used to specify the GP system that the filters are set up for. If the system attribute is set up for a filter group, the filtergroup will only apply to that specific GP system, and therefore will be ignored during auto-population from any other GP system.

Filter

Each filter group can contain one or more filter elements. Each filter will add to the restriction. The property attribute can be set to any node within the clinical information section. For example, for the conditions section, a filter can be set to any child element name of a condition. The comparison and value attributes are used in the same way as they are in GPASS auto-population.

Below an example for the past conditions table:

```
<question text="Pre-existing Conditions" type="dynaTable" id="pastcondition" object="Conditions">
    <column text="Condition Name" id="conditionname" mandatory="true" property="conditionname"/>
    <column text="Laterality" id="laterality" property="laterality"/>
    <column text="Modifier" id="modifier" property="modifier"/>
    <column text="Extension" id="comment" property="comment"/>
    <column type="Date" text="Date of Onset" id="date" property="date"/>
    <filters type="Generic">
        <filtergroup scheme="Read" system="EMIS">
            <filter property="Code" comparison="pattern" value="[Z1..]...."/>
            <filter property="EndDate" comparison="days" value="0"/>
        </filtergroup>
        <filtergroup scheme="Read" system="EMIS">
            <filter property="Code" comparison="pattern" value="[1-3]...."/>
            <filter property="EndDate" comparison="days" value="0"/>
        </filtergroup>
        <filtergroup scheme="SNOMED">
            <filter property="Code" comparison="pattern" value="[1-3]...."/>
            <filter property="EndDate" comparison="days" value="0"/>
        </filtergroup>
    </filters>
</question>
```

4.8. Regular Expressions

Regular Expressions (RegEx) are simple patterns of characters that can be used to evaluate strings. They date back to the work of an American mathematician by the name of Stephen Kleene (one of the most influential figures in the development of theoretical computer science) who developed regular expressions as a notation for describing what he called "the algebra of regular sets".

In a regular expression, everything is a generalized pattern. If I type the words "SCI Gateway" into my editor, I've created one instance of the words " SCI Gateway." If, however, I have a way to indicate to my editing software that I'm now typing a regular expression, I am in effect creating a template that matches all instances of the characters "S”,”C”,”I”,” ”,”G”,”a”,”t”,”e”,”w”,”a” and ”y" all in a row.

To construct more complex patterns we would use more generalised characters (called “meta characters”). There are a number of these meta characters in use, some of which are described below:

Meta Character	Description
\	Marks the next character as either a special character or a literal. For example, "n" matches the character "n". "\n" matches a newline character. The sequence "\\" matches "\" and "\(" matches "(".
^	Matches the beginning of input.
$	Matches the end of input.
*	Matches the preceding character zero or more times. For example, "zo*" matches either "z" or "zoo".
+	Matches the preceding character one or more times. For example, "zo+" matches "zoo" but not "z".
?	Matches the preceding character zero or one time. For example, "a?ve?" matches the "ve" in "never".
.	Matches any single character except a newline character.
(pattern)	Matches pattern and remembers the match. The matched substring can be retrieved from the resulting Matches collection, using Item [0]...[n]. To match parentheses characters ( ), use "\(" or "\)".
x|y	Matches either x or y. For example, "z|wood" matches "z" or "wood". "(z|w)oo" matches "zoo" or "wood".
{n}	n is a nonnegative integer. Matches exactly n times. For example, "o{2}" does not match the "o" in "Bob," but matches the first two o's in "foooood".
{n,}	n is a nonnegative integer. Matches at least n times. For example, "o{2,}" does not match the "o" in "Bob" and matches all the o's in "foooood." "o{1,}" is equivalent to "o+". "o{0,}" is equivalent to "o*".
{n,m}	m and n are nonnegative integers. Matches at least n and at most m times. For example, "o{1,3}" matches the first three o's in "fooooood." "o{0,1}" is equivalent to "o?".
[xyz]	A character set. Matches any one of the enclosed characters. For example, "[abc]" matches the "a" in "plain".
[^xyz]	A negative character set. Matches any character not enclosed. For example, "[^abc]" matches the "p" in "plain".
[a-z]	A range of characters. Matches any character in the specified range. For example, "[a-z]" matches any lowercase alphabetic character in the range "a" through "z".
[^m-z]	A negative range characters. Matches any character not in the specified range. For example, "[^m-z]" matches any character not in the range "m"

	through "z".
\b	Matches a word boundary, that is, the position between a word and a space. For example, "er\b" matches the "er" in "never" but not the "er" in "verb".
\B	Matches a non-word boundary. "ea*r\B" matches the "ear" in "never early".
\d	Matches a digit character. Equivalent to [0-9].
\D	Matches a non-digit character. Equivalent to [^0-9].
\f	Matches a form-feed character.
\n	Matches a newline character.
\r	Matches a carriage return character.
\s	Matches any white space including space, tab, form-feed, etc. Equivalent to "[\f\n\r\t\v]".
\S	Matches any nonwhite space character. Equivalent to "[^ \f\n\r\t\v]".
\t	Matches a tab character.
\v	Matches a vertical tab character.
\w	Matches any word character including underscore. Equivalent to "[A-Za-z0-9_]".
\W	Matches any non-word character. Equivalent to "[^A-Za-z0-9_]".
\num	Matches num, where num is a positive integer. A reference back to remembered matches. For example, "(.)\1" matches two consecutive identical characters.
\n	Matches n, where n is an octal escape value. Octal escape values must be 1, 2, or 3 digits long. For example, "\11" and "\011" both match a tab character. "\0011" is the equivalent of "\001" & "1". Octal escape values must not exceed 256. If they do, only the first two digits comprise the expression. Allows ASCII codes to be used in regular expressions.
\xn	Matches n, where n is a hexadecimal escape value. Hexadecimal escape values must be exactly two digits long. For example, "\x41" matches "A". "\x041" is equivalent to "\x04" & "1". Allows ASCII codes to be used in regular expressions.

Combining these meta characters with standard alphanumeric characters allows us create sophisticated patterns for matching.

4.8.1. Bracket Expressions and Sub Expressions

A Bracket Expression allows us to match ranges of characters. For example:

[0-9A-Za-z]

would match any single character that was a lower or uppercase letter, or a single digit number.

A sub expression allows us to store a part of a matched pattern for later reuse. When parenthesis is placed around a regular expression pattern, the pattern is stored within a buffer. This allows us to easily find occurrences of two identical words within a string of text. For example, look at the following sentence:

The balloon went up up in the sky.

As written, the sentence has a problem with repeated words. It would be nice to be able to devise a way to fix the sentence without having to look for duplicates of every single word. We can do this easily with the following RegEx:

\b([a-z]+) \1\b

The subexpression, in this case, is everything between parentheses. The captured expression includes one or more alphabetic characters, as specified by '[a-z]+'. The second part of the regular expression is the reference to the previously captured submatch, that is, the second occurrence of the word just matched by the parenthetical expression. '\1' is used to specify the first submatch. The word boundary Meta characters ensure that only separate words are detected. If they weren't, a phrase such as "is issued" or "this is" would be incorrectly identified by this expression.

4.8.2. Examples of RegExs

This RegEx will match a valid email address (in the form name@domain.ext):
^[\w-]+(?:\.[\w-]+)*@(?:[\w-]+\.)+[a-zA-Z]{2,7}$

This RegEx will match a UK format date (e.g. D/M/YYYY where D and M are 1 or 2 digits):
^\d{1,2}\/\d{1,2}\/\d{4}$

4.8.3. More Information

For more information please consult Microsoft Developer Network documentation on regular expressions:

http://msdn.microsoft.com/library/en-us/script56/html/reconRegularExpressions.asp

5. Appendix A – GPASS Enumerations

The following is a list of the current objects and parameters that can be included within protocols to extract data from the GPASS primary care system.

These are the values that will be inserted into the “objectOld” and “propertyOld” attributes of the `<question>` element.

Note. All enumerations are case sensitive.

key:

*	Properties which are custom items supported by the Gateway in addition to standard Open GPASS API properties
	Properties for planned future support which currently will not return data

“object” values	“property” values	Type	Description
Patient	Key	String	The unique key for the object. Generally of the form Number.Practice Code.Classname.
	AcceptanceDate	Date	Date of acceptance into practice.
	AcceptanceType	Unknown Birth First Acceptance Transfer In Immigrant Ex-services	Indicates the patient's status when accepted by practice.
	Address	String	Combined street address for the patient.
	Address1	String	Generally used for house name or omitted. Must be given if no Address2.
	Address2	String	Generally used for house number and street name. Must be given if no Address1.
	Address3	String	Generally used for a locality name or omitted.
	Address4	String	Generally used for Post Town.
	Address5	String	Generally used for county name or omitted.

	AgeInMonths	Long	Patient age in months
	AgeInWeeks	Long	Patient age in weeks
	AgeInYears	Long	Patient age in years
	BirthSurname	String	Surname patient born with
	BloodPressure*	String	“Systolic / Diastolic”
	BodyMassIndex*	double	Weight / (Height x Height). Negative indicates no value taken
	CHINumber	double	Patient CHI number.
	ConfirmationDate	Date	The date the practice received confirmation of the patient's registration.
	ContactPhone	string	Contact phone number for patient.
	DateOfBirth	Date	Date of birth.
	DateOfEnlistmentJoining	Date	Date of enlistment into services.
	DateOfEnlistmentLeaving	Date	Date of leaving services.
	Diastolic	Long	Diastolic blood pressure. Negative indicates no value taken.
	FootpathMiles	double	Miles on foot to patient's home.
	Forename	string	Patient's forename and initials.
	FullName	string	Patient's full name, surname first (eg. 'Smith, John B').
	Gravida	long	The number of pregnancies. Negative indicates no value taken.
	HealthBoardCipher	string	Patient's responsible HB cipher
	Height	Float	The height of the patient in metres. Negative indicates no value taken.

	HomeAddress1	string	Home address of a temporary resident, generally house name.
	HomeAddress2	string	Home address of a temporary resident, generally house number and street name.
	HomeAddress3	string	Home address of a temporary resident, generally locality name.
	HomeAddress4	string	Home address of a temporary resident, generally post town.
	HomeAddress5	string	Home address of a temporary resident, generally county name.
	HomeGPName	string	For a temporary resident, their registered GP's name.
	HomePostcode	string	Home address of a temporary resident, postcode.
	IsBornInUK	boolean	Indicates the patient was registered as born in the UK.
	IsDeducted	boolean	Indicates the patient is deducted.
	IsDeductionPending	boolean	Indicates a deduction request has been received from the health board, but that it has not been processed by the practice
	IsDispensing	boolean	Indicates the practice dispenses to the patient

	IsFrozen	boolean	Indicates the patient record is frozen pending transactions from the health board
	IsNewPatient	boolean	Indicates patient is new to practice and acceptance has not been approved by the health board
	IsOrganDonor	boolean	Indicates the patient has made an organ donor declaration
	IsPrivate	boolean	Indicates the patient is a private patient
	IsRecordRemovalPending	boolean	Indicates the health board has requested the records be sent following the deduction of the patient
	IsRecordSent	boolean	Indicates records have been sent by health board but not marked as received at the practice
	IsServiceDepandant	boolean	Indicates patient is a service depandant
	IsStayLessThan15	boolean	For a temporary patient, indicates the stay is for less then 15 days
	MaritalStatus	Unknown Single Married Divorced	Indicates marital status
	NHSNumber	string	Patient's NHS number, often omitted
	OccupationDescription	string	The Read term for the patient's occupation
	OccupationReadCode	string	The Read code for the patient's occupation
	OldAreaCode	string	Code (HB or FHSA cipher) for area patient previously registered in

	Parity	long	The parity (live births) of the patient. Negative indicates no value taken
	PatientID	string	Cross reference to practice filing system, not generated or processed by New Gpass
	Phone	string	Patient's phone number
	PlaceOfBirth	string	Patient's place of birth
	Postcode	string	Patient's postcode, formatted with a single space between parts
	PreviousAddress1	string	First line of address for patient in their previous HB - doesn't change if address changes within HB
	PreviousAddress2	string	Second line of address for patient in their previous HB - doesn't change if address changes within HB.
	PreviousAddress3	string	Third line of address for patient in their previous HB - doesn't change if address changes within HB
	PreviousCHINumber	double	Previously recorded CHI number
	PreviousDateOfBirth	date	Previously recorded date of birth
	PreviousGPName	string	Name of patient's previous GP from other practice
	PreviousHBCipher	string	Cipher of previous Scottish HB, if any
	PreviousNHSNumber	string	A previous number, if any

	PreviousSurname	string	Patient's previous surname, if any
	ReasonForDeduction	long	Indicates the deduction reason as per PARTNERS.
	RecordsReceivedDate	date	The date the practice received the patient's paper records
	RegisteredClinician	Clinician	The formally registered GP
	RoadMiles	long	Number of miles form patient home to practice using roadways
	SeanByClinician	Clinician	The GP who usually sees the patient
	ServiceCode	Unknown Permanent Temporary Resident Maternity only Child Health Surveillance only Immediately Necessary Treatment only Emergency Treatment only Contraceptive Services only Appointments only	Indicates whether the patient is registered on the practice list (permanent) or is a temporary patient of some sort
	ServiceNumber	string	For Ex-services patients, their service number
	Sex	Unknown Male Female	The patient's sex
	ShortAddress	string	House name, number and street name
	Surname	string	Patient's surname
	Systolic	long	Systolic blood pressure. Negative indicates no value taken
	Title	String	Patient's title
	UKEntryDate	date	Date of entry into UK.
	UKEntryDate2	date	Date of re-entry into UK
	UKLeavingDate	date	Date of leaving UK

	WaterMiles	double	Miles by water to patient's home
	Weight	float	The weight of the patient in kilograms. Negative indicates no value taken
Referral	Key	string	The unique key for the object. Generally of the form Number.Practice Code.Classname
	AppointmentDate	date	The date of the appointment made for the referral
	AttendanceType	Unknown First Visit Second Visit Subsequent Visit	Whether first, second or subsequent visit.
	AuthorisingClinician	Clinician	The clinician formally authorising the referral. Might not be the clinican conducting the encounter with the patient
	AuthorisingClinicianName*	string	Name of the Referring Clinician - linked to AuthorisingClinician
	Comment	string	Any additional information about the referral
	ConsultantForename *	string	The given name of the consultant being referred to
	ConsultantGrade *	string	The grade of the consultant (consultant / nurse)
	ConsultantID *	string	The ISD identifier for the consultant
	ConsultantSurname *	string	The surname of the consultant being referred to

	Date	date	Date the event was recorded, or date the prescription was last issued
	DateTreatmentCompleted	date	Final outcome date of referral
	DateTreatmentStarted	date	Date actual treatment commenced for referral
	ExpectedTreatmentDate	date	Date treatment is expected to start
	HospitalAddress *	string	The hospital address
	HospitalID *	string	The ISD identifier for the hospital
	HospitalName *	string	The hospital name
	HospitalTelephone *	string	The telephone contact number for the hospital
	Nature	string	Nature of Referral.
	Priority	Low/Routine Medium/Soon High/Urgent	Priority of the referral.
	Provider	string	Nature of Referral.
	Reason	string	Description of the ReasonReadCode
	ReasonReadCode	string	Read code for the reason
	ReferralType	string	The type of the referral.
	ReferredTo	string	The referral institution type.
	Speciality	string	The speciality of the referral.
	Specialty *	string	Equivalent to Speciality
RegisteredClinician	Key	string	The unique key for the object. Generally of the form Number.Practice Code.Classname
	GMCNumber *	string	The official GP's GMC code – retrieved from GPASS.
	GPCode		The official GP code, blank for non-GP Clinicians.

	IsActive	boolean	Indicates this clinician is currently working at the practice
	IsGP	boolean	Indicates if this is a registered GP (rather than a nurse, for example).
	Name	string	The full name of the clinician.
	Sex	Unknown Male Female	The clinician's sex
ReferringClinician	Key	string	The unique key for the object. Generally of the form Number.Practice Code.Classname
	GMCNumber*	string	The official GP's GMC code – retrieved from GPASS.
	GPCode	string	The official GP code, blank for non-GP Clinicians
	IsActive	boolean	Indicates this clinician is currently working at the practice
	IsGP	boolean	Indicates if this is a registered GP (rather than a nurse, for example)
	Name	string	The full name of the clinician
	Sex	Unknown Male Female	The clinician's sex
Premises	Key	string	The unique key for the object. Generally of the form Number.Practice Code.Classname
	Address *	string	The whole premises address.
	Address1	string	First line of premises address.
	Address2	string	Second line of premises address.

	Address3	string	Third line of premises address.
	Address4	string	Fourth line of premises address.
	Address5	string	Fifth line of premises address.
	Fax	string	Fax number of premise
	IsClinicalPremises	boolean	Whether or not the premise is actually used for clinical purposes. i.e. whether or not the premise can have surgeries assigned to it.
	Modem	string	Modem number for practice
	Name	string	Premises name
	Telephone *	string	Premises telephone number
	Telephone1	string	Premises main telephone number.
	Telephone2	string	Premises second telephone number
ReadCodeEvent	Key	string	The unique key for the object. Generally of the form Number.Practice Code.Classname
	Date	date	Date the event was recorded, or date the prescription was last issued
	EndDate	date	Date that indicates end of the condition.
	PlannedReviewDate	date	RFA data that indicates planned review of the condition
	Priority	2 - Low/Routine 1 - Medium/Soon 0 - High/Urgent	Indicates whether high, medium or low priority
	ReadCode	string	The encoded readcode for this event
	ReadCodeDescription	string	Textual description of the Read code

	ReadCodeMeaningExtension	string	Some free text about the event
	ReadCodeModifier	string	Modifier for the Read code.
	StartDate	date	Date that indicates start of the condition
PrescribedDrug	Key	string	The unique key for the object. Generally of the form Number.Practice Code.Classname
	Date	date	Date the event was recorded, or date the prescription was last issued
	BNFCode	string	A reference to the British National Formulary (BNF) section in which the drug appears. Format is 'nn.nn.nn.n'
	ConsultationInterval	long	Long integer specifying in number of weeks when the consultation is due
	DateStarted	date	The date the drug was first prescribed. This will be accurate for Gpass R5 repeat drugs - for acute and Gpass R4 this will be equal to the Date
	DatesIssuedCount	long	For repeat drugs, the number of dates when this prescription has been issued. This property is only available from Release 5 onwards
	Dose	string	String to contain the dose. Mandatory
	DrugName	string	String containing the name of the drug

	Frequency	string	String containing the frequency
	IsActive	boolean	To indicate if this repeat prescription is inactive or not
	IsAcute	boolean	Indicates this is a one-off acute prescription, rather than a repeat
	Message	string	String containing the advice text
	Preparation	string	String containing the preparation name
	PrescriptionInterval	long	Integer specifying in number of days how often a prescription can be issued
	Quantity	string	String containing the quantity
	TotalCost	double	Total cost in pounds of the prescription (unit cost * quantity)
ScreeningEvent	Key	string	The unique key for the object. Generally of the form Number.Practice Code.Classname
	Date	date	Date the event was recorded, or date the prescription was last issued
	ActionText	string	The description of the action to be taken
	Diastolic	long	Diastolic BP. Negative indicates no value taken
	Gravida	long	The number of pregnancies. Negative indicates no value taken
	Height	float	The height of the patient in metres. Negative indicates no value taken

	Location	In GP care Outwith GP care Unknown	The location where the screening measurement was carried out
	Parity	long	The parity (live births) of the patient. Negative indicates no value taken
	ProtocolName	string	The name of the protocol associated with the measurement
	RecallLetterDate	date	Date the recall letter was sent
	RecallLetterStatus	string	The text description of the letter status
	RepeatInterval	long	The time after which the screening should be repeated
	RepeatIntervalUnits	Unknown Weeks Months Years	The units in which the repeat interval are measured Weeks, Months, Years
	ResultLetterDate	date	Date the result letter was sent
	ResultLetterStatus	string	The text description of the letter status
	ResultReadCode1	string	The primary result read code
	ResultReadCode1Description	string	The term for the primary read code
	ResultReadCode2	string	Second read code required for cervical cytology
	ResultReadCode2Description	string	The term for the secondary read code
	Systolic	long	The systolic BP. Negative indicates no value taken
	Weight	float	The weight of the patient in kilograms. Negative indicates no value taken

6. Appendix B – Other GP Systems Enumerations

Patient information

object	property	
Patient	Chi	
	Key	
	DateOfBirth	
	Title	
	Forename	
	Surname	
	BirthSurname	
	Middlename	
	PreviousSurname	This is only for tertiary referrals. Third party systems do not contain this value, so cannot pre-populate it.
	Address	
	Postcode	
	Phone	
	MobilePhone	
	Email	
	Sex	
	MaritalStatus	
	HealthBoardCipher	
	Alcohol	
	Smoking	
	ExerciseStatus	
	NonTheraputicDrugs	
	ReligiousObservance	
	Diastolic	
	Systolic	
	Height	
	Weight	
	BodyMassIndex	
	Gravida	
	Parity	

Registered Clinician information

object	property	
RegisteredClinician	GMCNumber	
	GPCode	
	Name	
RegGpApplication	PracticeCode	
RegGpPremises	Name	
RegGpPremises	Address	
RegGpPremises	Telephone1	
RegGpPremises	Fax	
RegGpPremises	Email	

Referring clinician information

object	property	
ReferringClinician	GMCNumber	
	GPCode	
	Name	

RefGpApplication	PracticeCode	
RefGpPremises	Name	
RefGpPremises	Address	
RefGpPremises	Telephone1	
RefGpPremises	Fax	
RefGpPremises	Email	

Social circumstances information

object	property	
SocialCircumstances	Housing	
	Employment	
	Religion	
	EthnicOrigin	
	Language	

Clinical information

Allergies

object	property	
Allergies	Comment	
	RecordedDate	
	StartDate	
	Description	

Intolerances

object	property	
Intolerances	Comment	
	RecordedDate	
	StartDate	
	Description	

Risk to others

object	property	
RiskToOthers	RecordedDate	
	StartDate	
	Description	

Smoking consumption

object	property	
SmokingConsumptions	Description	
	Comment	
	RecordedDate	
	StartDate	

Alcohol consumption

object	property	
AlcoholConsumptions	Description	
	Comment	
	RecordedDate	
	StartDate	

Exercise status

object	property	

ExerciseStatuses	Description	
	Comment	
	RecordedDate	
	StartDate	

Conditions

object	property	
Conditions	ConditionName	
	Laterality	
	Modifier	
	StartDate	
	EndDate	
	RecordedDate	
	DateAssessed	
	ReviewDate	
	Comment	
	Priority	

Procedures

object	property	
Procedures	ProcedureName	
	Laterality	
	Modifier	
	DatePerformed	
	RecordedDate	
	Comment	
	Priority	

Family conditions

object	property	
FamilyCondition	ConditionName	
	Laterality	
	Modifier	
	StartDate	
	EndDate	
	RecordedDate	
	DateAssessed	
	ReviewDate	
	Comment	
	Priority	
	Relation	

Current medication

object	property	
PrescribedDrug	BnfCode	
	DrugName	
	Preparation	
	Dose	
	Quantity	
	Frequency	
	StartDate	
	EndDate	
	RecordedDate	
	LastIssuedDate	

Past medication

object	property	
PastMedication	BnfCode	
	DrugName	
	Preparation	
	Dose	
	Quantity	
	Frequency	
	StartDate	
	EndDate	
	RecordedDate	
	LastIssuedDate	

7. Appendix C – Predefined “id” Values

The following is a list of currently supported “id” values. This list is divided into three types of value.

- **Pre-defined Values** - For simple one-to-one mappings between questions and a data requirement of an NHS XML schema for clinical communications.
- **Pre-defined Complex Type Values** - Where a one-to-many mapping exists. For example where specific processing is required to handle medication/drug information.
- **Prefixed Generic Values** – Values that must be prefixed to allow protocol-specific data to be captured.

Note. All enumerations are case sensitive.

7.1. Pre-defined Values

The SCI Gateway uses named values for validation purposes. Gateway currently validates chi, postcode, date of birth, regGp_gmc_code, regGp_practice_Code, refGp_gmc_code, regGp_practice_Code, refhcp_scheme, refhcp_gmc_code, refhcp_forename, refhcp_surname, reforg_description, reforg_location_code, reforg_name, reforg_address, reforg_postcode and refhcp_postcode fields. The protocol verification tool will inform the user of what mandatory fields are missing. In the table below, the mandatory fields at schema level are indicated by (m-s), mandatory fields at application level by (m-a).

Patient Demographics	Registered GP Data	Referring GP Data
chi (m-a)	regGp_gmc_code	refGp_gmc_code
patient_dob (m-a)	regGp_practice_Code	refGp_practice_Code
patient_surname (m-a)	regGp_code	refGp_code
patient_forename (m-a)	regGp_name	refGp_name
patient_prev_surname		
patient_birth_surname		
patient_addressLine	regGp_practiceName	refGp_practiceName
patient_postCode	regGp_practice_addressLine	refGp_practice_addressLine
patient_phone	regGp_practice_phone	refGp_practice_phone
patient_mobile	regGp_practice_fax	refGp_practice_fax
patient_email	regGp_practice_email	refGp_practice_email
patient_orig_key		
patient_sex (m-a)		
marital_status		
health_board_cipher		
health_board		

Referring HCP Data
refhcp_scheme
refhcp_gmc_code
refhcp_forename
refhcp_surname
reforg_description
reforg_location_code
reforg_name
reforg_address
reforg_postcode

Referral Data	Consultant Data	Hospital Data
Date	HCP_ID	Hospital_ID
referral_date (m-s)	HCP_fullname	Hospital_Name
Protocol	HCP_forename	Hospital_Type
Specialty	HCP_surname	Hospital_Address
ReferralType (m-s)	HCP_grade	Hospital_Phone
referral_expectedOutcome (m-s)		

Main Condition Data	Priority	Transport Instructions
mainCondition_readcode (m-s)	Priority (m-s)	transport_instructions
mainCondition_description (m-s)	Priority_reason	transport_required
mainCondition_OnsetDate		transport_type
mainCondition_Comment		transport_datetime
mainCondition_Laterality		transport_pickup
mainCondition_Modifier		transport_destination
mainCondition_Certainty		
maincondition2_ readcode		
maincondition2_description		
maincondition2_onsetdate		
maincondition2_comment		
maincondition2_laterality		
maincondition2_modifier		
maincondition2_certainty		
maincondition3_ readcode		
maincondition3_description		
maincondition3_onsetdate		
maincondition3_comment		
maincondition3_laterality		
maincondition3_modifier		
maincondition3_certainty		

Risks and Alerts	Social Circumstances	Patient Consent and Sensitivity
RiskAlcohol	socialCirc_Housing	patient_consents_to_share
RiskSmoking	socialCirc_Employment	document_sensitivity
RiskExercise	socialCirc_Religion	
riskNonTherapeuticDrug	socialCirc_EthnicOrigin	
religiousObservance	socialCirc_Language	

7.2. Pre-defined Complex Type Values

The following “id” values are used to handle patient and clinical data that may vary in size depending on the nature of the clinical communication. As such any protocol `<question>` element that is required to capture this information must be of type “dynaTable” to allow multiple sets of data to be captured. In all cases, the attributes “name” and “id” defined within the `<column>` element must not be changed.

Single dynatables

These dynatables can only be used once within the protocol.

Value	Description

PastCondition	Patient past conditions
PastProcedure	Patient past procedures
FamilyCondition	Patient's family history
administeredMedication	Current administered medication
PastMedication	Historic administered medication
alert_allergy	Patient allergies
alert_intolerances	Patient intolerances
alert_risktoothers	Patient's potential risks to others

Group dynatables

Dynatables prefixed with "examination_", "investigation_" and "admin_" are group dynatables. This means that multiple tables can be defined with these prefixes.

Value	Description
Examination_[Identifier]	Examinations
Investigation_[Identifier]	Investigations
Admin_[Identifier]	Administrative information

Note. When including these types of question within a protocol it is recommended that the question be copied from an existing protocol, to ensure consistency and identical behaviour between protocols.

Columns

The following tables can be used to verify the correctness of the column identifiers specified for the above tables and dynatables. All date columns should be specified with a "type" attribute that has a value of "date".

'Past medication', 'Administered medication'	Pre-population
Bnfcode	bnfcode
Drugname (mandatory)	drugname
Preparation	preparation
Dose	dose
Frequency	frequency
Startdate	startdate
Enddate	enddate

Note: No other columns are allowed.

'alert_allergy', 'alert_intolerances', 'alert_risktoothers'	Pre-population
Comment	readcodemeaningextension
description	readcodedescription
startdate	startdate
recordeddate	date
modifier	readcodemodifier

Note: No other columns are allowed.

'Past condition'	Pre-population
Conditionname (mandatory)	readcodedescription
Laterality	-
Modifier	readcodemodifier
Comment	readcodemeaningextension

Date	startdate OR date
recordeddate	date

Note: No other columns are allowed.

'Past procedures'	Pre-population
Procedurename (mandatory)	readcodedescription
Laterality	-
Modifier	readcodemodifier
Date	startdate OR date
recordeddate	date
comment	readcodemeaningextension

Note: No other columns are allowed.

'Family Condition'	Pre-population
Conditionname (mandatory)	readcodedescription
Laterality	-
Modifier	readcodemodifier
Date	startdate OR date
Comment	readcodemeaningextension
Relation	-
recordeddate	date

'Examination_', 'Investigation_'	Pre-population
Description (mandatory)	-
Result	-
Date	-

For dynatables prefixed with 'Admin_', any columns can be defined because the information in each row is added to the message XML as one pbr:AdditionalInfo node.

Note: No other columns are allowed.

7.3. Prefixed Generic Values

Value	Description	Example
admin_	For administrative questions	"admin_wheelchair_user"
investigation_	For clinical investigation questions	"investigation_abdominal_pain"
examination_	For clinical examination questions	"examination_weight"
verification_	For verification a protocol page has been visited	"verification_clinicaldata"

8. Appendix D - Protocol Specification Template

8.1. Purpose of Document

To provide detail of a new Protocol being added to SCI Gateway, including which data should be extracted and supplied to external systems.

8.2. Specification

8.3. Purpose of Protocol

Some background information on the requirement for the protocol, including the intended usage/users and whether this is an update to an existing protocol or a completely new protocol.

8.4. Data to be Captured From GPASS

A list of the data required to be captured from GPASS and which field it should be displayed in within the protocol.

8.5. Data to be Captured From Third Party GP System

A list of the data required to be captured from the third party GP system and which field it should be displayed in within the protocol.

8.6. Data to be Captured From Protocol

Detail of the data that should be supplied from the Protocol questions, including where and how it should appear.

8.7. Data to be Displayed in Gateway Letter

From the detail captured above, what should be displayed and where in the resulting letter. This is especially relevant as some letters have a standard format and length, meaning that not all the data captured previously will be able to be displayed.

8.8. Referral Data to be Displayed in SCI Outpatients

For referrals, when all data has been captured from GPASS or the third party GP system and the Protocol, the user can use SCI Gateway to book an appointment if the destination hospital has SCI Applications. Once the appointment date and time have been picked, SCI Gateway will automatically book an appointment and transfer the data to SCI Outpatients. However, Outpatients controls the content of the referral letter displayed and hence is different to that displayed in the Gateway referral letter. This section should specify which data should be displayed in Outpatients.

9. Appendix E – SCI Gateway Protocol Form

The form on the next page should be used for sending protocols to the SCI Gateway Support Team.

Scottish Care Information

Information required for Creating/Editing Protocols

Trust/Board	
Hospital/Location	
Region	
Contact Name of Person who has created/edited the protocol	
Name of Protocol	
Version	
Release No:	
Date posted to SCI	
Date Protocol Goes Live in Training/Live (5 working days from receipt)	
Instance e.g. Training/Live or both	
New/Existing Protocol	
Region	
Address	
Tele No	
Email address	
Name of Person in Medical Records authorising protocol	
Contact Details of Medical Records	

Hospital/Locations the protocol has to be applied to:

Hospital/Location	Specialty	Consultant	Protocol

When finished please send completed form to the SCI Team at

gpasshelpdeskcalls@gpass.csa.scot.nhs.uk

Additional Information: