OpenEyes - Contacts

Editors: G W Aylward

Version: 0.9:

Date issued: 16 June 2010

Target Audience

General Interest	✔
Healthcare managers	✔
Ophthalmologists	✔
Developers	✔

Amendment Record

Issue	Description	Author	Date
0.9	First issue	G W Aylward	16 June 2010

Table of Contents

Introduction

The need to communicate between healthcare professionals about the care of patients means that OpenEyes requires a system of managing professional contacts. Several types of health professional might be involved in the care of ophthalmic patients, including Consultants, General practitioners, Optometrists, Orthoptists, and Ophthalmic Nurses.

The Contacts Table

All contacts within OpenEyes are stored in a single table, and distinguished by a type field. The fields are as follows;

Field	Type	Comments
contact_id	INT UNSIGNED NOT NULL AUTO_INCREMENT	Primary Key, 4 billion
paskey	VARCHAR(12)	Primary key in PAS
practitioner_code	CHAR(8)	CfH code as in CfH supplied data
title	VARCHAR(8)	
first_name	VARCHAR(20) NOT NULL	
last_name	VARCHAR(40) NOT NULL	
degrees	VARCHAR(20)	
nick_name	VARCHAR(20)	
description	VARCHAR(40)	
company	VARCHAR(80)	
address1	VARCHAR(40)	
address2	VARCHAR(40)	
city	VARCHAR(24)	
postcode	VARCHAR(8)	
country	VARCHAR(16)	
telephone	VARCHAR(24)	
fax	VARCHAR(24)	
email	VARCHAR(60)	

Field	Type	Comments
type	ENUM("Consultant", "GP", "Optometrist", "Specialist", "Solicitor", "Other", "Social Worker", "Health Visitor", "Other")	

Managing the data

Populating and maintaining the contacts table is essential to aid the user in communication with referring doctors and other health professionals, but this is not a trivial task. The approach adopted for OpenEyes is to initially populate the contacts table with the nest sources of information that can be found, and then maintain the data as the need arises.

Initial population of data

There are a number of sources of information with which to populate the contacts table. Connecting for Health offers a list of consultants categorised by specialty and organisation. With knowledge of the specialty codes, and a list of NHS organisations, this information can be used to produce a list of names and addresses of Consultant Ophthalmologists in England and Wales (See Appendix 1). Similar lists may exist for optometric practices, and some other professional groups. GP data is recorded on the PAS, and can be taken from there when the patient is registered.

Initial inspection of the CfH list indicates it to be accurate and contemporaneous. It included 75% of the current Moorfields consultants, but of the 25% absent, these were all either joint appointments, or those appointed within the previous 6 months.

Table 2. Sources of contact information

Contact type	Source	Comment
Consultant Ophthalmologist	CfH	Adequate coverage
Other consultants	CfH	Adequate coverage
General Practitioners	PAS	
Optometrists	?	
Orthoptists	?	
Social Workers	?	
Health Visitors	?	

Maintenance of data

The receipt of a referral letter on a patient represents an opportunity to update of add details of contacts. The following flow chart illustrates the sequence of actions when correspondence is received on a patient.

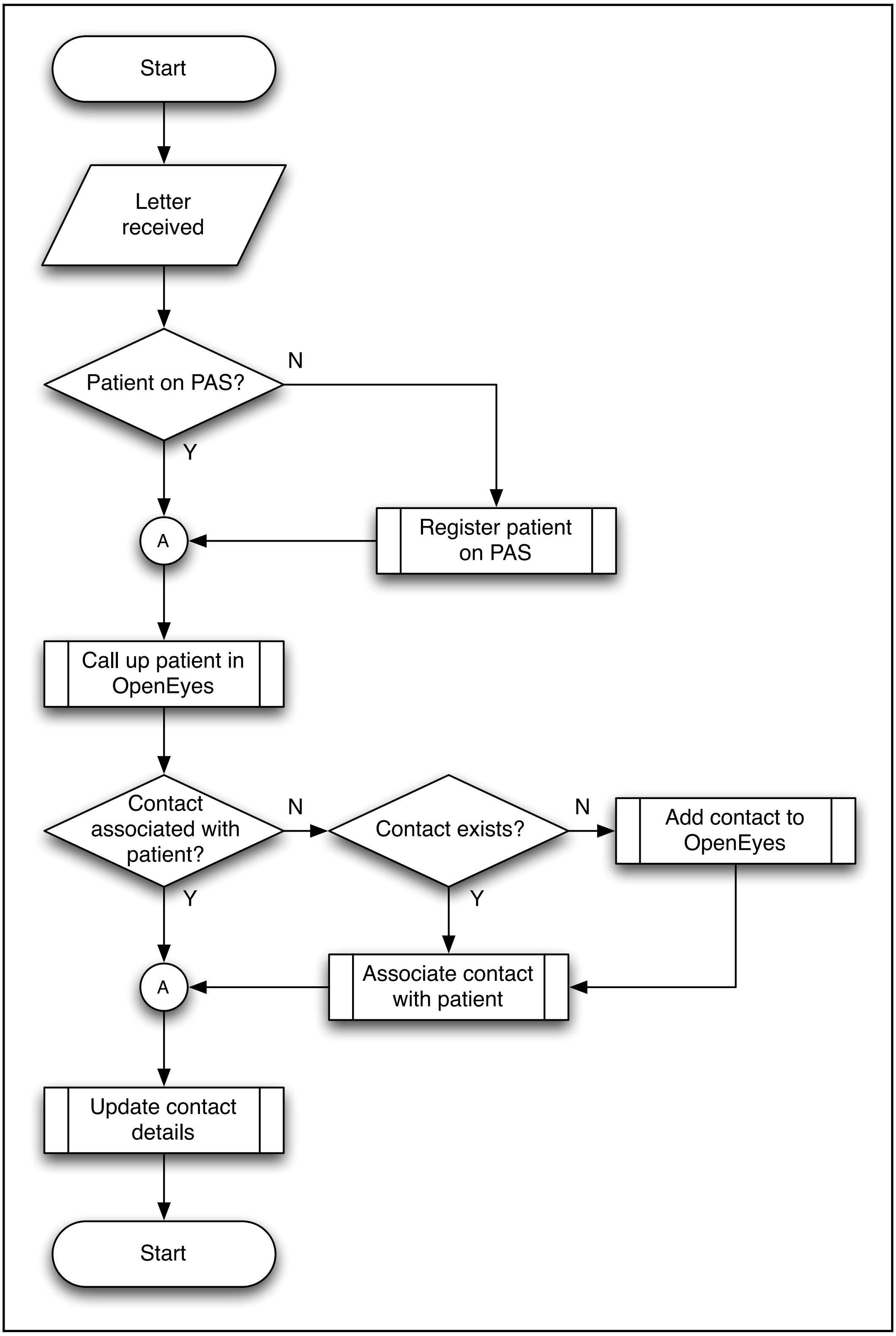

Figure 1. Flow chart illustrating process for maintaining contact database from incoming referral letters

Appendix 1

Extracting data from the CfH website into the contacts table

The following procedure will import into OpenEyes the current list of NHS consultants provided by Connecting for Health.

1. Acquire the data

Download the list as a comma separated file from the CfH website (N3 access required).

2. Create a database

Create a new database to import the data into with the following SQL statements;

```
‣ CREATE DATABASE consultants;
‣ USE consultants;
```

3. Create a consultants table

Create a table to import the consultant data into with the following SQl statement;

```
‣ CREATE TABLE consultants (
  consultant_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
  gmc_number CHAR(7),
  practitioner_code CHAR(8),
  surname VARCHAR(75),
  initials CHAR(3),
  sex CHAR(1),
  specialty_code SMALLINT UNSIGNED,
  practitioner_type CHAR(1),
  organisation_code CHAR(5),
  PRIMARY KEY (consultant_id)
  );
```

4. Import the data

Read the ‘econcur.csv’ file into the consultants table using SQL management software, or the following statement;

```
‣ LOAD DATA INFILE.. (TODO: Test statement)
```

5. Create an organisation table

Create a table to import the organisation data into with the following SQl statement;

```
‣ CREATE TABLE organisations (
  organisation_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
  organisation_code CHAR(5),
  trust VARCHAR(40),
  address1 VARCHAR(40),
  address2 VARCHAR(40),
  city VARCHAR(24),
  postcode VARCHAR(8),
  PRIMARY KEY (organisation_id)
  )
```

6. Import the data

Read the ‘ETR.csv’ file into the consultants table using SQL management software, or the following statement;

```
▸ LOAD DATA INFILE.. (TODO: Test statement)
```

7. Deal with Case

The organisation name and address is all in upper case text, which is not suitable for use in communications. The following MySQL script will create a stored procedure in the consultants database for converting to sentence, or ‘proper’ case.

```
delimiter //
DROP PROCEDURE IF EXISTS propercase//
CREATE FUNCTION propercase(strIn varchar(100)) RETURNS varchar(100)
BEGIN

SET @strTmp = LCASE(strIn);

SET @strOut = '';

SET @i = 0;
WHILE @i < LENGTH(@strTmp) DO
    BEGIN
        SET @c = SUBSTRING( @strTmp, @i, 1 );

        IF @i < 1 THEN
            SET @c = UCASE(@c);
        ELSE
            BEGIN
                SET @lc = SUBSTRING( @strTmp, @i - 1, 1 );
                IF @lc = ' ' THEN
                    SET @c = UCASE(@c);
                END IF;
            END;
        END IF;


        SET @strOut = CONCAT(@strOut, @c);
        SET @i = @i + 1;
    END;
END WHILE;

SET @strOut = REPLACE(@strOut, 'Nhs', 'NHS');
```

```
RETURN @strOut;
END;
//
delimiter ;
```

8. Combine data

Use the following select to combine data from both tables.

```
SELECT c.practitioner_code AS practitioner_code,
IF(c.sex = 'M', 'Mr', 'Miss') AS title,
c.initials AS first_name,
c.surname AS last_name,
'Consultant Ophthalmologist' AS description,
propercase(o.trust) AS company,
propercase(o.address1) AS address1,
propercase(o.address2) AS address2,
propercase(o.city) AS city,
o.postcode AS postcode
FROM consultants AS c
INNER JOIN organisations AS o USING(organisation_code)
WHERE c.specialty_code = 130
```

9. Clean the data

Manually inspect for any errors, (such as MCCloud instead of McCloud) and correct them.

10.Export to a CSV file

The data can now be exported to a .csv file for importation into the OpenEyes contact database