[1]CLIMSOFT vers 3.2 to vers 4 Data Model

Written by
Myles Kirk-Gushowaty, Incisive Services Group
&
Cathy Garlick, Statistical Services Centre

Comments:
Dr. Rafael Posada, SASSCAL/DWD
(on behalf of the Climsoft Technical Advisory Group)

December 2014

Comments from:

Technical Advisory Group (TAG)

Cathy Garlick?

Steve Palmer?

Rafael Posada

Jens Riede

Track of changes refers to the comments made during the Meeting in MetOffice in December 2014

[1]CLIMSOFT is a Climate Data Management System written by Samuel Machua & Albert Mhanda with development supported by Kenya Meteorological Service, Zimbabwe Meteorological Service, ACMAD, UK Met Office and the World Meteorological Organisation.

Content

General Remarks arose during the Climsoft Technical Workshop

This section points out the main issues arose during the Climsoft Technical Workshop in Nairobi (30th July to 1st August 2014). It is divided in three different subsections. The first one refers to the concerns arose during the data model discussion. The second one refers to the issues addressed while discussing the CDMS Recommendations from the WMO. The last one shows the new proposed E/R Diagram for v4 from Albert Mhanda.

Data model Discussion

- *There will be only one database built in MySQL, instead of three in MS-Access AGREED*
- *Common Agreement on that a backup of Climsoft db in an ftp and also in their own computer is needed (make sure that the people care about making backups of their data) This can be handled as a "business rule"*
- *Make the MySQL Workbench available for the administrators of Climsoft (not available for the users). This tool allows making backups. AGREED*
- *Climsoft v4 should be an "easy-to-use" software. The documentation is very important, but it should be easy to start using Climsoft without going through all the documentation (it should be "intuitive") AGREED*
- *It was proposed to link the "station_element" and "observation" entities in the way that observations can only be introduced when the "station_element" is defined. Otherwise, it is not possible to introduce observation values. This is to ensure that, for instance, a station with no thermometer will be linked to temperature observations). The developers want to look at this issue in detail and try it. AGREED TO BE DONE FOR V.4*
- *When having data from different data sources (e.g. GHCN, FAO, CDIAC, etc.), it is necessary to store the name of the source, when and how the data were accessed, etc. It was suggested to use the entity "acquisition" for this purpose. The "acquisition_type" attribute of entity "observation" will be used as a foreign key that links to the "acquisition" entity. This entity will then include the name of the source, the data of accessing the data, and the origin of the data (e.g. ftp address). AGREED to use attribute "adquisition_type" in "observation" to store this information. A descriptive table "acquisition" will be set up in V.4 but will not be linked to the "observation" table*
- *The Regional Climate Centres should develop training material based in the WMO Recommendations / Documents. This should be part of the "Demonstration Phase" of the RCC. OUT OF SCOPE OF THE MEETING*

CDMS Recommendations

- *Data Dictionary (Section 4.2.1.1) → not a priority, but keep it in mind. There is something done on that in the ECA&D (http://eca.knmi.nl/dailydata/datadictionary.php) OUT OF SCOPE OF THE MEETING*
- *Data processing (Section 4.3.1.8) → this should be kept in mind, but it is not a priority (it also might be covered by the new entities related to "instrument")*

- *Network (Section 4.3.1.10) → Station qualifier in the "station" entity. However, this does not allow having more than one network for a single station. Therefore, the network will be implemented in the "stationClass". A guidance to the NMHS has to be given in relation to the Network AGREED to be covered by the table "station_qualifier"*
- *Data Discovery metadata (Section 4.3.2) → it should de outside the data model. It is necessary to identify someone that has done that already and share it through the WIS. OUT OF SCOPE OF THE MEETING*
- *Business Rules (Section 5.1.1.1) & WMO messages (Section 5.1.1.2) → outside of the data model. They can be considered as "plugins" The Business Rules will be taken into account in V.4. NOT CLEAR if WMO messages (e.g BUFR, SYNOP, etc.) will be created using "plugs-in".*
- *Status log (Section 5.1.1.6) → this can be covered within other logs as plugin. Further discussion on this issue is needed. This is already working in V3.2. Will be maintained for V.4*
- *Heuristic checks (Section 5.3.1.3) → not included in the data model. OUT OF SCOPE OF THE DATA MODEL // It depends on the experience of the Operator. Furthermore, Climsoft does not make any QC changes automatically, but is the user who has to do it manually.*
- *Data governance, controlled access to data and systems (Section 3.2.1.1) → improve the roles already defined in Climsoft. Give guidance to the decision-makers of the met services in this issue. WILL BE SOLVED when using MySQL. The roles discussed are: Admin, Observer, Key-entry, QC, Station Manager and Enquires.*
- *Managed change (Section 3.2.2.2) OUT OF SCOPE OF THE DATA MODEL // It has to be solved by the NHMS. A System Management guideline / Best Practices document in this issue is recommended*
- *Standard normal of the WMO (Section 4.4) → Including them in an entity within the Climsoft db to store the WMO standards in the db. How should the entity look like? AGREED to create a new entity including the WMO standards and estimations. The management of these standards should not be done by the Administrator but by someone else. Rules on the calculation of these products in the WMO documents (see Section 4.4)*
- *Discovery metadata catalog (8.2.1.1) → The ECA&D already has this kind of information. It can be used as a template NEEDED IN THE FUTURE (Example is the "Diagram of station data against time" created by Rwanda Meteorology Agency, http://www.meteorwanda.gov.rw/index.php?id=55)*
- *Scheduling (Section 9.2.1.1) → Need to make additional documentation on this issue (including for ingestion, products and backup) Already available in V.3.2 under AWS Real Time Interface. The Backups will be considered as a product saved manually by the user.*

New Data model proposed

See Annex 1

Personal Remarks

In this section few comments are pointed out, although they were not discussed during the Technical Workshop:

- *Maintain consistency in naming entities and attributes. When they are defined by more than one name I would split the names with "underscore" (_) rather than writing them all together. I would write the names always in lower-case.*
  Example: "station_element" instead of "stationElement" Remove of "underscore".
- *Make sure that all the date/time attributes are of type "DATE/TIME" instead of "TEXT" (the attribute "begin_datetime" from the "lookup_location table of the temp db is TEXT, whereas the data type of "begin_datetime" from the "station_location" table of the intermediate db is "DATE/TIME" AGREED. All date/time fields will be saved as type "date/time"*

Introduction

This document describes the data model used in the databases for CLIMSOFT version 3.2. The main entities and the relationships between them will be described. This document also includes suggestions for possible changes to the model.

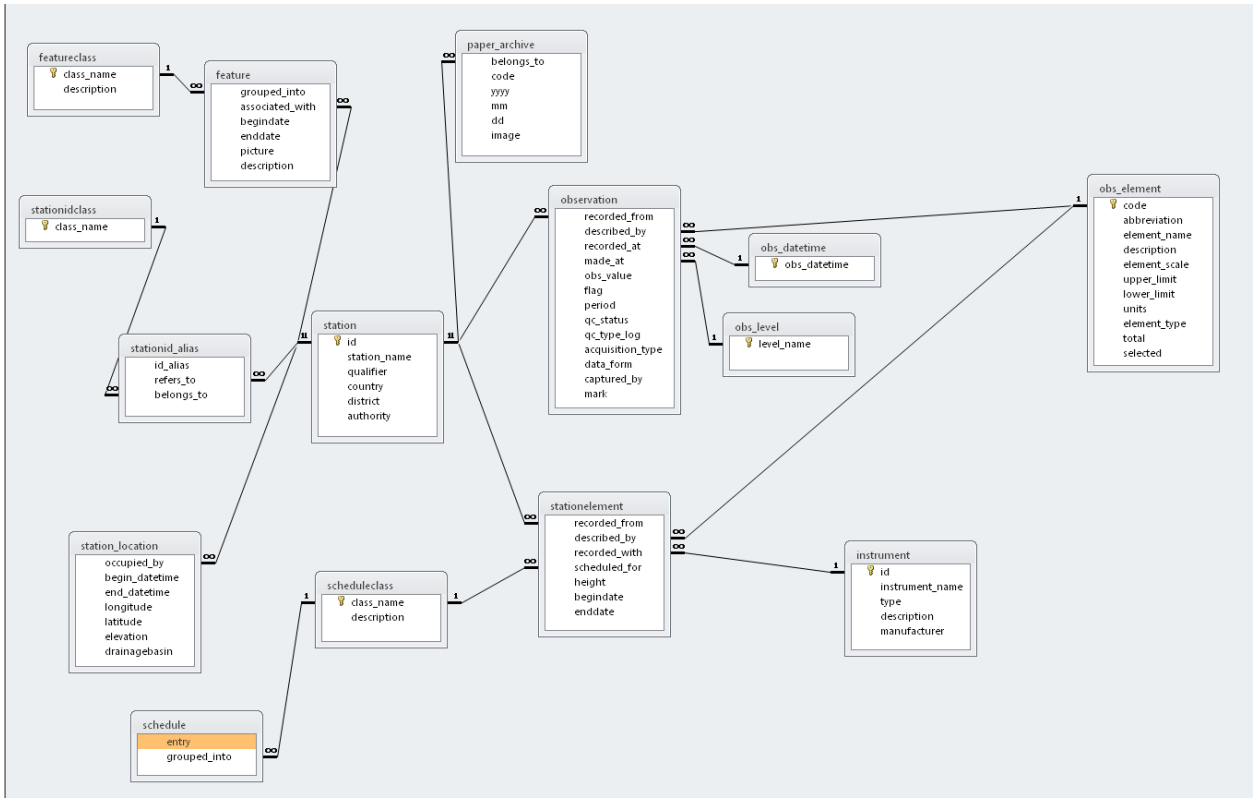
CLIMSOFT Databases

In the CLIMSOFT system there are currently three databases: a temporary database into which data are originally entered by the user; an intermediate database which has a more normalised structure and into which data are transferred for quality control checks; and a main database to which data are transferred once checks have been performed. The structures of the intermediate and main databases are similar. This document describes the structure of the main database.

The Data Model

The entity-relationship diagram for the intermediate database is shown in Figure 1.

Figure 1 - Entity-Relationship Diagram for CLIMSOFT Database



The model is based around the observation with each recorded value being a record in the **observation** table. Each observation is defined by what it represents **(obs_element)**, where it was recorded **(station)** and when it was recorded **(obs_datetime)**. The **observation** table can be thought of as being at the "lowest level" of the structure; it stores the smallest item of data; i.e. the actual measured value.

At the other end of the spectrum we have the **station** table and from this table we can link to all the data recorded at each station.

This model gives a great deal of flexibility as different stations may record different elements at different times and the structure allows for other elements to be added easily.

We'll now go through the entities (tables) in the model in turn, describing the attributes and properties of each one.

Database Entities

observation: Stores observation values and related information such as where it was recorded, what it represents and the time it was recorded.

TAG: *there will be two "observation" tables in v4: one "observationInitial" (raw data) and "observationFinal" (quality controlled data). This is to keep the original observed values in case they are corrected after QC. This structure will be tested during the development process.*

Jens: I would prefer "observation_initial" for raw data and "observation_proved" for controlled data. AGREED on keeping the names "observationInitial" & "observationFinal"

AGREED on two observation tables: observationInitial (no QC values) & observationFinal (QC values).

SUGGESTED to remove those values that have gone through QC process and have not been modified from the table "observationInitial". This will be offered as a Business Rule (the user will decide whether to remove those values or not from the "observationInitial")

station: Stores station information such as name, unique identifier, country, etc.

obs_element: Stores information about meteorological elements including upper and lower limits, unit of measurement, etc.

TAG: *This entity has around 200 elements. It is proposed to keep only the "parameters" in the "obs_element" entity (temperature, precip, etc.) and to use the attribute "acquire_type" of the "observation" entity to specify where the data comes from (e.g from the AWS).*

AGREED on keeping an meaningful name of the element

AGREED on removing commas and semicolons from element names and descriptions. Use underscore instead.

obs_datetime:

Date and time when an observation was made.

TAG: *"obs_datetime" will be removed as entity in v4. It will be included as attribute in entity "observation"*

AGREED on removing this entity

obs_level: Level at which the observation was made. This could be surface level or a standard pressure level such as 850. The unit of measurement here is HectoPascals (formally millibars); these are pressure heights used for radio-sonde observations.

TAG: *"obs_level" will be removed as entity in v4. It will be included as attribute in entity "observation"*

AGREED on removing this entity

AGREED on creating a new table for upper air obserations (e.g. "uAirObs"). The "obs_level" attribute won't be represented in the observation entities

AGREED on further research on how to design the new entity

Example given by Steve Palmer:

uAObs
recordedFrom describedBy uAObsDateTime obsLevel obsLocationLat obsLocationLon QCStatus QCFlag

station_location:

Captures information on station history; i.e. change of geographical location with time.

station_element:

Stores information about the measurement of particular elements at particular stations. This might include information of when an element started being recorded at that station, the instrument used, etc.

feature: Stores information about physical features associated with stations. For example: a large tree close to the station. This table is not used in the current version of CLIMSOFT.

TAG: *This entity could be used to record the operator of the weather station. This is an open issue. This can be considered as a "Business Rule"*

WMO Recommendations: *The "feature" entity has to cover the "local environment" of the station (section: 4.3.1.6 → Local environment) SOLVED by using the new entity "physicalFeature" (see Figure 6)*

WMO Recommendations: *The "feature" entity has to include an attribute including the WMO index for the convenience of an installation site (Section: 5.4.1.1 → Siting Classification). Maybe it should be placed in another entity, since the "siting class" applies to each sensor, rather than to a station as a whole (maybe a new "attribute" in the "instrument" entity?!) SOLVED by using the attribute "description" in the entity "physicalFeature"*

WMO Recommendations: *New attribute in the "feature" entity that covers the user restrictions for a specific station (Section: 4.3.1.2 → Station overview) AGREED on creating a restriction attribute in the "StationLocationHistory". E.g. as "VARCHAR". If it is NULL, the station would have no restrictions (free available)*

WMO Recommendations: *Concerning "Time zone" (Section 4.3.1.5 → Location), a new attribute in the "feature" entity has to be included to store whether the station is set up in local time or in UTC. The key entry should also have the option to set up this variable. The "observation" entity will ALWAYS save the observation values in UTC. AGREED on removing the "timeZone" from the "observationInitial" Entity. AGREED on that the "observation" entities will always save the time in UTC.*

In order to make sure that the observations are entered in UTC, it was suggested that a new entity with all the time zones has to be created. The downloadable files of http://timezonedb.com/download can be used as template.

SUGGESTED that the entity "stationQualifier" will be related to the new entity through the attributes "stationTimeZone" and "dataSourceTimeZone".

feature_class:

Groupings of features. Again this table is not used in the current version of CLIMSOFT.

stationid_alias:

Secondary identifier for a station. This is used when a station is referred to using different identifiers by different organisations. For example there might be a specific WMO ID for a station.

stationid_class:

This is effectively a lookup table which stores the different types of station identifier as a group or class of identifiers.

schedule: This stores a set of observation times ranging from 00:00 (midnight) through to 23:00 (11pm).

scheduleclass:

This stores names assigned to an observation schedule. For example *synoptic1* would be an observation every three hours between 06:00 and 15:00.

instrument: This contains information about instruments used for station elements.

TAG: *New "instrument" entities such as: "instrumentInspection", "instrumentFaultReport" and "faultResolution" to meet the recommendations of the WMO (monitoring metadata) were created for version 4. This is still an open issue to discuss.*

COVER in the new Database Diagram (see Figure 6)

WMO Recommendations: *This entity, together with the ones created by the developers, has to cover the "Sensor" metadata (section 4.3.1.7 → Sensor). (e.g. the new attribute: "serialNumber" in the "instrument" entity)*

COVER in the new Database Diagram (see Figure 6)

WMO Recommendations: *Concerning the uncertainties of the instruments, the WMO recommends keeping the information of the processes, software, governance mechanisms and data analysis used to understand and record the uncertainty inherent in observation measurements and processes. This will be covered by an attribute at the "instrument" entity. (numeric field according to the type of uncertainty?) (Section 5.4.4.2 → Measurements)*

COVER to some extent in the new Database Diagram (see Figure 6)

WMO Recommendations: *The new entity "instrumentInspection" covers the status of the sensor through the attribute "conditionOK". Change the name of the attribute to "status"*

and including several options on it, such as "operational", "not operational", etc. (Section 4.3.17 → Sensor)

Observation

The **observation** table has the following attributes (fields):

recorded_from:

This is the link to the **station** table and represents the "where" part of the definition.

described_by:

This is the link to the **obs_element** table and as such describes what the value represents; e.g. temperature, precipitation, etc.

recorded_at:

This links to the **obs_datetime** table and records the time and date that the observation was made.

obs_value: This is the value itself; it is a numeric field stored as fixed format with one decimal place.

made_at: This is a link to the **obs_level** table which records whether the observation was made at surface level or at some other level.
AGREED to be removed

flag: The flag represents whether the value is missing, trace, estimated or generated and if used should be one of "M", "T", "E", or "G".

*In the database structure there is a table called **flags** but there is no link from this field to this table; perhaps there should be. Also this field has a maximum of 50 characters but if we are expecting just a single character then this could be reduced to 1.–*

One character is enough – but see comments on qc_status below. I don't think "Missing" is useful, since a missing value is not present in the observation table.

*Personally, I would rather see this replaced with **Version** with possible values of 0 or 1 where Version=0 means that the observation value is "Originally received before amendment" and Version=1 means "best QC'd value" If one did this, then **Version** would need to be a primary key element. This would allow both the original and the amended value to exist in the database. Most observation records would have Version=1 and no Version=0. Normal enquiries would only use Version=1 but research or legal enquires might want to get Version=0 as well. There is no need to store intermediate values arising at different stages of the QC process.*

Strictly one could have a "Trace" value which is also "Estimated" so these should be separated.

"Generated" is a more difficult concept – there is an argument that derived data should not be stored in the same database as the core observations because of the difficulty of ensuring consistency. On the other hand, on-demand generation of commonly-used values such as monthly rainfall imposes an overhead.

Considering that there is no programming language choosen for the upcoming version 4 /5/…I would prefer a flag base on 1 (1= "Originally received before amendment") and two for qc_proofed data. To avoid problems with the code of further version!!

AGREED on keeping the flags "M", "T", "E" and "G". Use of "M" only for Key-Entry

AGREED on having a reference entity describing the flags, but not related to other entities of the db.

AGREED on that "no flag" will mean "original value"

AGREED on that the attribute "flag" will be of type "VARCHAR(1)"

AGREED on that the Documentation of Climsoft should include which equations/formulas have been used to generate values (e.g. relative humidity)

NOT SOLVED how to record the equations/formulas that have been used by the NHMS to generate a value

period: This field allows for data values that are accumulated over a period and is often used for rainfall, maximum and minimum temperatures. This field allows for irregular data or missing data. For example if a daily rainfall reading is missed, the next one would cover two or even three days. The **period** would then be 48 hours or 72 hours for a Monday after a weekend gap.

*Other than this being a number field I don't know what this represents – is it needed? Yes, though it is not properly used at present – this should be fully implemented in Version 4. This allows for data types which are accumulated over a period, especially rainfall, max and min temperature. Usually these are recorded on a regular schedule (which is covered in the schedule and schedule class tables) but this allows for irregular data or missing data. In particular, if a daily rainfall reading is missed, the next one would cover two or three days. This is a valid reading, but the **period** is then 48 hours (or 72 hours for the Monday after a weekend gap). If this accumulation does not span a month-end, then the accumulated monthly rainfall is made up of valid readings (though if it does span the month-end, then the monthly accumulation would strictly be an estimate).*

AGREED on using this attribute to specify the time period used to calculate the observed value (e.g. 24, 72, etc.)

SUGGESTED to be saved in hours

qc_status: This field is for recording the level of quality control at the level of the observation value.

*The intermediate database has a table called **qc_status_definition** but it is empty. Should there be some codes and descriptions in here and should this table also be in the main database?*

(version 4?) This should be in the main database, and store flags showing the QC history of the observation record. One flag could be for TRACE (this only applies to manual rainfall observations). Other flags would record which QC operations have been performed, they would also record the acquisition type (since this affects the relevant QC operations), and also the reason code if an estimated value has been produced. This could use a binary code in a

single variable.
AGREED on the current three levels of QC:
0 → No QC check
1 → QC check and no changes made
2 → QC check and value modified

qc_type_log:

This is for recording a log of quality control operations carried out on this observation.

I would combine this with **qc_status**.
AGREED on calling this attribute as "**qcTypeId**"

acquisition_type:

This indicates how the data were entered and the options are:
1. Key-entry
2. CLICOM
3. CSV import
4. AWS (Automatic Weather Station)

*There is a table called **acquisition_type** in the intermediate database but not in the main database. I think this table should also be in the main database and there should be a relationship from this field to this table.*

*I would combine this with **qc_status**.*
SUGGESTED to have a reference entity describing the different acquisition types.
AGREED that this attribute will specify the SOURCE where the data come from (e.g. GHCN, SASSCAL, etc.)

data_form: This field stores the data entry form that was used for entering this observation value.

Is this automatically entered by the CLIMSOFT system? I notice it is an open-ended field and there is no related list of data entry forms – perhaps there should be?

Probably not worth storing in the Observation table. It should be covered in the Schedule table.
AGREED on having a reference table to describe the different forms available

captured_by:

This field is to record the user who entered the observation value.

Again is this entered automatically and does it just pick up the username from the login screen?

Again, probably not worth storing in the Observation table.
AGREED on that this information will be seen only by the administrator
AGREED on that the entry of this attribute should be the user nickname (e.g. operator1) rather than the real name (specify this as a business rule??)

The **observation** table does not have a primary key defined; however it does have a unique index consisting of the following fields:

- recorded_from
- described_by
- recorded_at
- made_at
- acquisition_type
- qc_status

Therefore each of these fields must have a value for every observation value recorded and the combination of the values in these fields must be unique.

*Is it necessary to have all these fields as part of the unique index? With the index as it currently is you could, for example, have several observation records for minimum temperature recorded at a particular time from a particular station but one entered directly through key entry and one entered via a .csv import. That to me is slightly worrying as I can see no way of determining which of these is the correct value. I would have thought the unique index would just need **recorded_from, described_by** and **recorded_at**; we would then have the "where", "what" and "when" and would not be able to have more than one observation of a particular element at the same time and place.*

*If it is essential that **made_at, acquisition_type** and **qc_status** should all have values, then their the **required** property on each of these fields can be set to **Yes** but I question whether they are needed as part of the unique index for the reasons stated above.*

*Yes, this structure is essential!!!!! (though see comments above on **acquisition_type** and **qc_status** and the possibility of adding **Version**). Also **observation_period** should be added to the compound Primary Key, especially for rainfall, max and min temperature.*

The compound Primary Key enforces referential integrity. As you note, it does allow two or more values of a single element from a station. A station may have several instruments (e.g. several raingauges) each of which provides valid observations. The climatologist would make a policy decision on which one to use for a particular purpose, but might want to use the others, especially to fill in gaps. In particular, both the AWS and manual observations are useful, but are likely to be different. (Being political, this is what the Met Service specialists are for, to understand and advise on the detail – all observations are approximations to some "true" value, but that "true" value is a noisy turbulence phenomenon).

SUGGESTED to create a Business Rule that specifies a list of proiorities to import the data into the observationFinal. This is to avoid that there will be more than one record for the same date/time, station and element. It was suggested that the highest priority should be for the values coming from a Key Entry.

Station

The **station** table has the following attributes:

id: Unique station identifier. This is the primary key field for this table and therefore must be unique and must always have a value.

*I wonder whether this should be renamed as **station_id** to be more consistent with how it is used.*
AGREED on using "stationId" instead of "id"

station_name:

The name of the station – generally entered in upper case.

qualifier:

*The qualifier seems to be to do with the type of station and what elements might be recorded. Example values for this field that I have come across include **SYNOPTIC** and **RAINFALL**. Are there a limited number of values for this field and if so perhaps these should be listed in a separate linked table to restrict what can be entered in this field.*

AGREED on having a new entity named "stationQualifier" (see Figure 6)

Country: This is the country where the station is located: e.g. KENYA.

District: The district where the station is located: e.g. MACHAKOS.

Authority: This records who owns and operates the station. This is particularly useful for co-operating observers who may be from other organisations. This is becoming more important as more automatic weather stations are installed by other organisations. These may not meet formal standards for exposure or instrumentation but can provide much useful information. This allows us to distinguish between "formal" and "informal" data.

Obs_element

This table stores information about each element that could be recorded at a typical meteorological station. This table has the following attributes:

Code: A unique code associated with the element – this is the primary key and so must be unique and must have a value.
AGREED on calling this attribute "elementId"

Abbreviation

An abbreviation for the element. This can take up to 50 characters but in the current list the maximum number of characters used is 12.

I think the abbreviation should be unique as I think this would make it easier when referring to a particular element. The current element list is over 600 and I'm sure there are several duplicates in this list.

This is a mess inherited from CLICOM, and should be simplified radically. In particular, many of these elements include the observation period as text e.g. Daily rainfall, Hourly rainfall. I argue that all rainfall total observations should be named "Rainfall", and then have a Period qualifier as well as the time of observation. While this sounds complicated to the end-user, it removes the possibility of confusion. Also I would remove any which have "AWS" in the title – the observation system should be recorded elsewhere.

The "abbreviation" attribute should be unique. For instance, the element "PRECIP" appears two times in the table: one corresponding to code 5 and the other to 104

Element_name:

The element name can take up to 100 characters but the maximum used in the current list is only 30. Some of these names are very cryptic.

Description: This is a longer description of the element and can take up to 200 characters but the maximum used is only 60.

I see very little difference between the element_name and the description and wonder whether we need both. My feeling is that we could drop the element_name and make the description more descriptive – currently for some elements this doesn't give an adequate or meaningful description.

There is a case for keeping Abbreviation, Name and Description, especially since there is work in WMO to define a naming convention which uses both full and abbreviated names. Agreed, the Description should be much more informative.

AGREED on having a better description of each element
AGREED on cleaning up the list of elements, but probably for future versions

Element_scale:

Many elements are recorded with a scale factor – for example temperature is often recorded with a scale factor of 0.1 – this is common meteorological practice and is designed to help avoid decimals in the data values. For example 10°C would be entered as 100 with a scale of 0.1.

Upper_limit:

Lower_limit:

These values represent the possible range for the element and this can be customised to suit local conditions.

Is there a reason why these fields are text fields in the database? Intuitively I would have expected these to be numeric fields.

Units: This is the unit of measurement for the element and this is a text field as would be expected.

I'm wondering whether the unit should be in a separate related table – there could then be a field in the observation table called something like "expressed_in" to link to the table of units of measurement. My feeling for instance is that temperature measured in degrees F is the same element as temperature measured in degrees C, the only difference is the unit of measurement so there should just be one element record for temperature.

Alternatively if you prefer to keep the unit together with the element and so have separate element codes for temperature in degrees F and temperature in degrees C there should perhaps be a lookup table with the list of possible units. This would help avoid spelling errors. For example in the current list we have ***Celcius,***

Deg Celcius, deg cent, deg Centigrade, Degrees, Degrees C, Degres, Degres C *all of which I suspect are the same.*

Agreed, this would be sensible.
AGREED on that the observations should be saved always in the same units. A tool should be able to convert the units, when the observations are entered in other units.

Element_type:

I can't see any major difference between ***units*** *and* ***element_type****. In over half the current records (319/620) the values in these two fields are identical; for 233 cases* ***element_type*** *is blank; and for the further 6* ***units*** *is blank. In only 58/620 cases is there any difference in these fields and for these 58 cases the value of* ***element_type*** *is either "minute", "hourly", "daily" or "monthly".*

Period should not be stored in this way – but this will need the addition of a Period attribute to the main Observations table.
SUGGESTED to use this "attribute" as a descriptor of "atmospheric", "terrestrial" or "oceanic", following the nomenclature given by the Essential Climate Variables (ECV) of GCOS.

Total: This is a boolean (Yes/No) field which determines whether or not totals will be calculated as a check.

Is this always the monthly total? Would it make sense to also use the mean?
AGREED on removing the attribute from the "ObsElement", since it is used for QC but it is not a qualifier of the observation.

Selected: This is another boolean field which is ticked if the element is selected for the station. By default 35 elements are selected. These are shown in Table 1
AGREED on removing the attribute from the "ObsElement", since it is not a qualifier of the observation.

Table 1 - Default Selected Elements

Element Code	Description
3	Temperature; daily minimum
4	Temperature; daily mean
5	Precipitation; daily total
18	Evaporation;pan1;daily total
84	Sunshine; Daily total
101	Temperature, dry bulb
104	Precipitation; total hourly
105	Relative humidity at 06Z
106	Pressure; station
112	Wind direction
115	Cloud opacity; total
543	Soil; Moisture daily at 10 cm
544	Soil; Moisture daily at 20 cm
545	Soil; Moisture daily at 30 cm
546	Soil; Moisture daily at 40 cm

549	Soil; Moisture daily at 50 cm
550	Soil; Moisture daily at 60 cm
881	AWS; Air Temperature
882	AWS; Maximum Temperature
883	AWS; Minimum Temperature
884	AWS; Barometric Pressure
885	AWS; Dew Point Temperature
886	AWS; Wind Gust Speed
887	AWS; Wind Gust Direction
888	AWS; Sunshine Hours
889	AWS; Station Level Pressure
890	AWS; Actual Sea Level Pressure
891	AWS; Standard Seas Level Pressure
892	AWS; Precipitation Cumulative
893	AWS; Relative Humidity
894	AWS; Solar Radiation
895	AWS; Wind Direction Average
896	AWS; Wind Direction Instantaneous
897	AWS; Wind Speed Average
898	AWS; Wind Speed Instantaneous

Obs_datetime

This table has a single field which is also called **obs_datetime**. This links to the **observation** table through the field **recorded_at** and contains the date and time the value was recorded.

*I'm wondering whether this table is needed in its current form. With only a single field it doesn't seem to be serving any purpose as the date value is repeated in the **observation** table anyway.*

This is a hang-over from Logbook, and also because in earlier versions, the Observation table had separate elements for the days, hours minutes etc. It makes much more sense just to use a Datetime variable in the Observations table and remove this table.
AGREED on removing this entity

Obs_level

This table only has one field which is called **level_name**. This is a text field and possible values are 500, 600, 700, 850, and "surface".

I'm assuming these refer to different heights above the ground/sea level; but there is no indication of unit of measurement and I'm not convinced of the use of this table in the current structure. Clearly we want to record that the observation was made at a particular level but there are some elements in the list that include the height (or depth) at which the measurement was made. So it seems we have a mixture of methods here and we have the odd situation where we could be recorded "soil temperature at 50cm" made at "850" which obviously doesn't make sense. My feeling is that we either include the height or depth in with the element and so have different elements for different levels, or we have the separate table for the level. Having the separate table is better in terms of normalisation of the data model, but having separate elements might be easier in practice. What we should move away from is the mix that we currently have.

This is a "business rule" – one can only record temperature, humidity, wind speed and direction and geopotential height at levels above the Surface. No need to build this rule into the database referential integrity.

AGREED on removing this entity

Station_location

There is a one-to-many link between the station and the station_location which allows for stations to change location over a period of time. The fields in this table are:

Occupied_by:

This is the link to the station ID

Begin_datetime:

Date when the station first started operating at this location – this is a Date/Time field.

End_datetime:

Date when the station stopped operating at this location – also a Date/Time field.

Is there anything in the CLIMSOFT code to check for station locations that overlap in time?

Probably not, but this is a business rule. One would want to run both sites in parallel for an overlap period, but usually this will be covered by renaming the old site if it remains after the changeover of primary site. This is a messy area!

Longitude: degrees longitude

Latitude: degrees latitude –

Both longitude and latitude are stored as text values – is there any reason why they are not stored as numeric? Is there a recommended format for GPS coordinates?

elevation: altitude above sea level.

Elevation is also stored as text – is there a reason for this? Also I'm assuming this is metres above sea level but there is no indication of the unit of measurement.

Should be metres above MSLP, and should refer to the key instrument. In many cases this will be the height AMSP of the ground at the location of the main daily raingauge.
AGREED on changing the field type of this attribute to "FLOAT"

Drainagebasin:

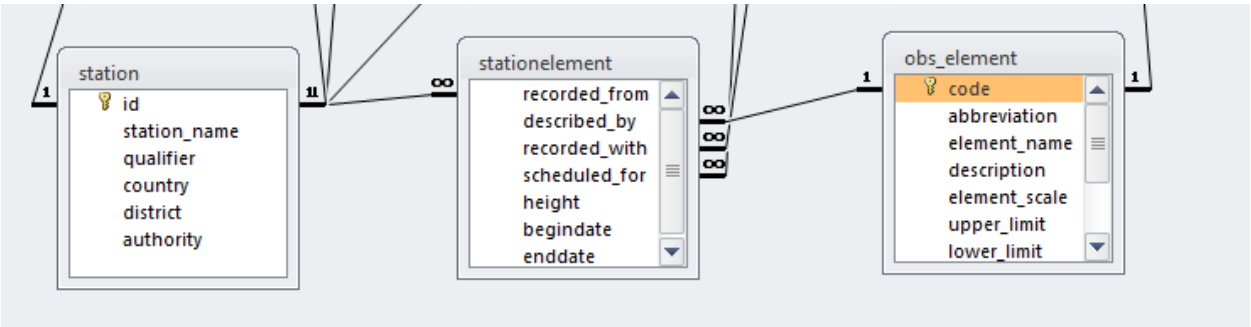
I have no idea what this is – please explain.

This links stations to river catchment areas (drainage basin) so that one can group the data.

Station_element

This table describes elements that are recorded at a particular station and is linked to the station table and to the obs_element table. This table can be thought of as a "junction" or "joining" table separating the many-to-many relationship that would otherwise exist between **Station** and **obs_element** where many elements are collected at each station and an element is collected at many stations. This relationship is shown in Figure 2~~Figure 4~~.

Figure 2 - Relationship between Station and Element



Recorded_from:

This is the link to the **Station** table and stores the Station ID.

Described_by:

This is the link to the **obs_element** table and stores the unique element code.

Recorded_with:

This links to a table of instruments. With the current data structure an instrument can be used to measure many station elements.

Scheduled_for:

This links to the **scheduleclass** table which gives the frequency of measurements.

Height: *Height above sea level? No unit of measurement are specified here – the field is a long integer so can take whole numbers. Is this field necessary as many elements include their height/depth in their descriptions – this seems to be another instance of mixing methods allowing for observation metadata that doesn't make sense.*

This should be the height of the instrument relative to the Station Elevation (see above) AGREED on removing this attribute from the stationElement entity. The height will be covered by the "instrument" entity.

Begindate: Date this element started to be collected at this station.

Enddate: Date element stopped being collected at this station.

Begindate and enddate are both stored as text fields – is there a reason for this?

Should be datetime variables, but this is a hangover. Note this is not the same as begin and end dates for the particular instrument or sensor.

There is no primary key defined for this table but there is a unique index consisting of the following attributes:

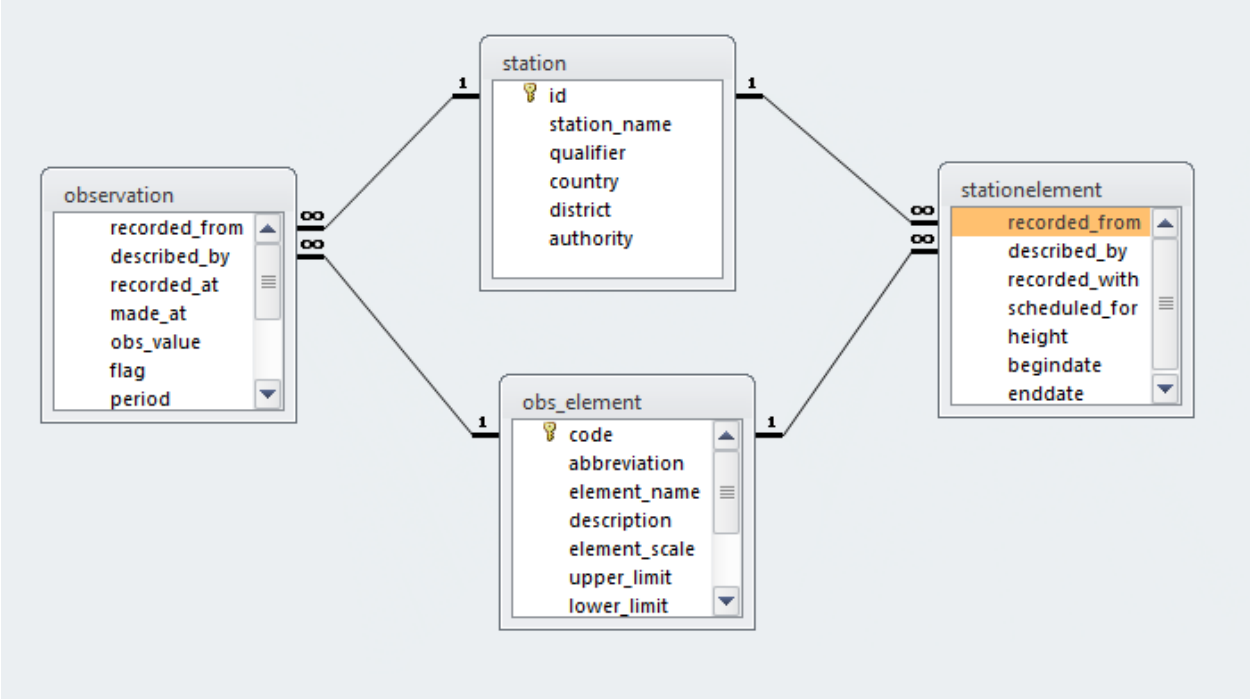
Recorded_from
described_by
scheduled_for
recorded_with
begindate

In other words that combination of values must be unique.

One thing that isn't clear to me is how this table links to the observations. Currently there is no link between ***stationelement*** *and* ***observation*** *which means observations can be entered for any element not just those that are specific to the current station. With the "what" "where" "when" idea of describing the observation, this* ***stationelement*** *table describes the "what" and the "where" and to make use of this implied restriction on elements I would propose linking* ***observation*** *to* ***stationelement*** *instead of linking it separately to both* ***station*** *and* ***obs_element****.*

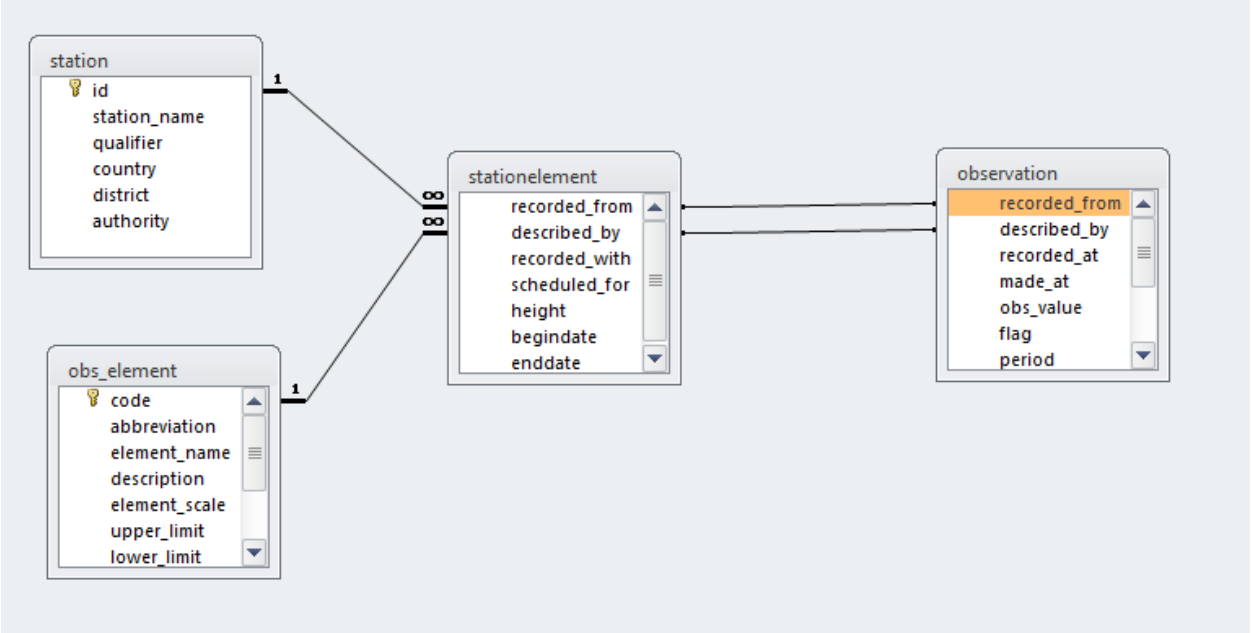
So instead of having this structure between these tables:

Figure 3 - Current structure of observation & Stationelement



We would have this structure:

Figure 4 -Proposed Structure



Some changes would be needed in the key identifiers in order to enforce referential integrity on this relationship, but unless I'm missing something I think this would be a stronger data structure in terms of restricting the observations at a station to be only for the elements selected at that station.

Agree completely! This would also sort out the mess of multiple element names.
AGREED on that the suggested relationship (Figure 4) is not possible in a real db diagram. Therefore a "Business Rule" will be needed to enforce that the "stationElement" exists before importing the observations for a given element into the observation entities.
AGREED on that, if the instrument is not known, then the user can still record the obs_value. The user will then save the information of the instrument as "unknown", which is already more than having the entity empty.

Feature

The **feature** table is linked to the **station** table so one station can have many features. The attributes are as follows:

Grouped_into

This is the link to the **station** table.

Associated_with:

This links to a table called **featureclass** which I assume is for grouping features in some way.

Begindate: The date the feature appeared?
AGREED on type "Date/time" instead of VARCHAR

Enddate: The date the feature disappeared?
AGREED on type "Date/time" instead of VARCHAR

How do you define the start and end of a feature – a tree for instance has probably been in a particular location for many hundreds of years and didn't just appear overnight? Also these fields are both text fields rather than Date/Time fields – is there any reason for this?

Also I wonder whether the naming of the two linking fields is the wrong way round – to me I would assume a feature is grouped into a feature class and associated with a station not the other way round.

Picture: This is where an image of the feature can be included in the database.

Description: A description of the feature – this is a memo field so can take up to 64,000 characters.

Are descriptions really that long or would a text field of 255 characters be large enough for the description of the feature.

This could store an inspectors report on the station, so needs to store a lot of text.

There is no primary key for this table but there is a unique identification index consisting of **grouped_into, associated_with** and **begindate**.

Featureclass

Features are grouped into feature classes. The primary key here is the **class_name** a text field of 50 characters maximum. There is one other field which is the **description** which is also a text field of 50 characters maximum.

Can you give some examples of feature classes as it's not entirely clear to me what sort of things you would expect in this table.

It could be e.g. Surroundings, Maintenance, or almost anything else which might be useful to know about the Station.

Stationid_alias

This table links to the **station** table and to the **stationidclass** table. It has the following fields:

Id_alias: This is an alternative identification by which the station might be known by other organisations. This is not set as the primary key but is set as a unique index which amounts to the same thing – for ease of understanding it could be set as the primary key. The only difference that would make would be to disallow blanks in this field.

Refers_to: This links to the **station** table and so is the station ID.

Belongs_to: This specifies that a particular alias belongs to a particular class. The link to the table **stationidclass** ensures that values in this field are restricted.

I would suggest that the id_alias can accept duplicates, since it can happen that there are two different stations with the same WMO identifier (wmo_id). This is the case when we have the same station, but from different sources. This station will have then two different "id", but will have the same "wmo_id".
AGREED on that organising the data from the different sources before entering them into the Climsoft observations.

Stationidclass

This table has a single field called **class_name** which is a text field and is the primary key. This is effectively used as a lookup table for the grouping of the station ID aliases. Records in this table by default are:

- iata_id
- icao_id
- local_id
- rainfall_id
- wmo_id

Perhaps a description field in this table would be useful for explaining what each of these classes refer to.
AGREED on including reference table with a description of each station Id Class

Schedule

This table links to the **scheduleclass** table and has just two fields which are:

entry:

grouped_into:

This is a link to the **scheduleclass** table and as such stores the name of the schedule class.

There is no primary key for this table but the combination of the two fields forms a unique identification index.

I'm assuming there are many "entries" in each schedule class and from the temporary database I'm assuming an entry is an hour in the day so if a schedule class is described as "every hour between 8am and 12pm" then I would assume there would be 5 records in the schedule table matching this – e.g. if this schedule class were called "Cathy" then there would be the following records in the schedule table:

Entry	***Grouped_into***
8	*Cathy*
9	*Cathy*
10	*Cathy*
11	*Cathy*
12	*Cathy*

*Is there any reason why the **entry** field is text rather than numeric – if it was called something like **hours** it would be clearer?*

It is not this complicated! The Schedule table just allows one to link one or more Scheduleclass values to a single station. So a station could have several Schedules (e.g. "Every hour 24/7" "06Z every day" "06Z – 15Z three-hourly Monday – Friday").

Scheduleclass

This table links to the **schedule** and to the **stationelement** tables. One schedule class record has many schedule records and one schedule class record have many station element records.

The fields are **class_name** and **description** both of which are text field with a maximum size of 50 characters.

*I can see the logic behind these tables and the links, observations for a particular station element would normally be scheduled for collection at particular times of the day as described by a specified scheduleclass, and the **schedule** table gives details of the hours that the observations would normally be made for that scheduleclass. What I'm not clear on though is how this scheduling fits in with when the observations are actually made – i.e. the **obs_datetime** link to the observation table.*

It is not intended to be fully prescriptive, it should be descriptive only.

Instrument

This table contains information about instruments used for station elements. It is linked to the **stationelement** table so that each instrument could be used for recording several elements. The table has the following attributes:

id: This is the unique identifier for the instrument and is the primary key.
AGREED on using "instrumentId" instead "id"

Instrument_name:
The name of the instrument.

In each of the three records in the main database this field is blank – I'm assuming it is the name of the instrument but not sure how this would differ from data in the other fields.

type: This is the type of instrument and current records are *Barometer, Rainguage, Thermometer.* This field is a unique index.

***Type** is a unique index but why is this the case – I would have thought there might be thermometers, etc. that perhaps had a different manufacturer but under current restrictions there can only be one record for thermometers.*

Description: A description of the instrument – current cases are *Fortin barometer, Standard rainguage, Drybulb thermometer*.

manufacturer:
Which organisation made the instrument – currently all records have *Casella* as the manufacturer.

All fields are text fields of length 50 which is probably not right particularly for the ID field which appears as though it should be numeric in the main database at least.

In the main database the contents of this table are:

id	instrument_name	type	description	manufactur	Click to Add
1		Barometer	Fortin barometer	Casella	
2		Rainguage	Standard Rainguage	Casella	
3		Thermometer	Drybulb Thermometer	Casella	

*But in the temporary database the contents of the **lookup_instrument** table are:*

lookup_instrument

code	instrument_name	instrument_type	description	manufacturer	Click
BARO_1	Barometer		Fortin barometer	Casella	
EVAP_I					
PRECIP_I	Rainguage		Standard Rainguage	Casella	
SODE_I					
TEMP_I	Thermometer		Drybulb Thermometer	Casella	
VISIB_I					
WIND_I					

Why is there a difference? The gaps in the instrument table in the temporary database are causing some problems – the code is not shown in the drop-down list for instruments and therefore blanks are showing in the list but their meaning is consequently hidden from the user. When entering the metadata for a station you can do silly things such as set the element to temperature but say the instrument for this element is a rainguage.

There is a major problem with this table – I think it should be split into two. One is for the particular instrument or sensor, including e.g. Manufacturer, model and type, Instrument_serial, Instrument_start, Instrument_end and Instrument_calibration_date and Instrument_calibration attributes (possibly with a further table to allow for multiple calibration events during the lifetime of the sensor, and with something about whether the instrument is currently in store or unserviceable – though this might be better in a separate Stores database….).

A further table is needed to link the particular instrument to the Station_element for the Station, again with start and end dates. This would record service replacement of sensors in the station equipment without a break in the Station_element. This would then allow one to drill down and find out which sensor was being used at a station at a particular time – climate change scientists want to know why there are jumps in a record, and this can be due to instrument drift.

Performance of Climsoft v3.2 (University of Kenya)

Comments given by James K. Musyoka, from the University of Kenya.

Metadata:

- The Metadata button appears at the bottom of the “Welcome Window” and is suggested to be more at the beginning. A suggested order of the buttons shown in the “Welcom Window” is:

User Admin → Metadata → Key Entry → DB Utilities → QC → Products → AWS Real Time

AGREED on that the order of the buttons will remain as it is in v3.2

Key Entry:

- When using the "data for one element for one month" to enter daily rainfall data, a totals mismatch error message was sometimes returned even when the total figure provided is correct.

  On clicking the "Help" button, Climsoft was not able to locate the help files on its own.

- Improve the description of the different forms. It could be like this:
  - Hourly data for one element
  - Daily data for one element
  - Daily data for several elements
  - Daily upper air data for several elements

AGREED on that the description needs to be improved

QC:

- All the CQ reports seem to work well as described in the guide with the exception of "View Neighbouring Values" which returned a "No current record" error message.

  AGREED on including the option of giving the radius in "km" besides degrees
  AGREED on improving the description of the error "No current record".

- The action of closing and reopening the QC dialogue resulted in an empty Intermediate db file box.
- This forces the user to manually browse for the db file in the climsoft folder under program files if they are to carry out any operations.
- It may help if the users are provided with certain graphs/plots that give an idea of the quality of the data, and it should be provided for each station separately.
  AGREED on giving the option of QC checks for each station separately
  AGREED on creating graphics for QC checks (using other language such as R)
- The R product for Climsoft could help with this if this is not possible to implement in Climsoft

Products

- Only the tabulated data and backup files were generated successfully from this section
- For the inventory, an error message about too few parameters was returned.
- For the graphs, an error message that no data has been selected even when this has been done was returned.
  The error will be checked
- To improve the option of "Daily for Instat" in "tabulated data", rename the column headings for the years as y'year', e.g. y2014. It would be neat if the data file retrieved from Climsoft does not need further editing for Instat.
  This will be done

- To improve the option “Daily Data, Transpose” in “tabulated data”:
    - Single column for the full date value (instead of, or in addition to the current three separate columns year, month,day).
    - The data file with many elements, output a column for each element, instead of putting together all the observed values in one single column

The year, month and day will remain separate as it will be more secure.

Climsoft Questionaire

Overview of Responses

Installation of the database

SOLVED: The new version in MySQL will allow making the installation process easier

User configuration

AGREED on strengthening the user password (e.g require a minimum of characters)

AGREED on giving a list of user

Key Entry Forms

AGREED on simplifying the name of the forms, using “daily”, “monthy”, “hourly”, etc. (e.g. “hourly data for one element”; “daily data for one element”, etc.

Function keys

AGREED on including a “search” tool in the Key-entry forms, so that the users can find the record of a specific station and date easily.

Sequencers

AGREED on creating a sequence interface so that the administrator can customize the sequencers

Sort functions

For user monitoring there is already the attribute “capturedBy” in the observation entities. This can be seen by the administrator, so that he/she can monitor the entries of each user.

Uploading data into the database

AGREED on creating a tool to be able to change the obsValues of the entity “observationFinal“ for the Administrator and the QC operators

AGREED on using a “tracking system” (e.g. “Track” or “Red mine”) as a help desk

Data ingestion and import

AGREED on making possible to import CLICOM data from .csv files that do not have “flag columns” after each observation value.

AGREED on creating a tool to import data from the NOAA-ftp

AGREED on making possible to see the folders and files located in servers (i.e. outside of the localhost) when the Windows Explorer pops up to select an importable file

Data extract and retrieving products

AGREED on making the user designate the name of the file before saving.

Data transmission and decoding

This section is not clear and needs more clarification

Products generation

AGREED on using R or python to improve products of Climsoft

AGREED on collaborating with University of Reading and the AMI Team in Kenya, who are currently working on creating R-codes to produce Climsoft-products

AGREED on implementing the message "No QC errors found" instead of "No QC data found"

What other successful products could Climsoft offer?

Quality Control

AGREED on improving the Select Dialog, so that it looks similar similar to the "Update Main Db"-option. It should include the possibility of:

- making QC for a station
- selecting more than one element at a time, and even all of them

Flags and meta data to improve the record validation process

SOLVED. The new version will improve the metadata of the Instruments

Other suggestions for improvement

AGREED on improving archiving procedure. There is something done in Climsoft V2 and it is related on how to proceed in order to successfully archive the data. Create a kind of Guidance.

AGREED on giving the option to the user to generate summaries as products (e.g. decadal values)

Wider issues of concern

Videos in French and Portuguese will be released. The ones in French will be available in March

AGREED on that there is a need of guidance on how to deal with climsoft (e.g. creating documentation specifying where and/or how the data are located). This may be specific for each NMHS

Open Issues in Climsoft v3.2

This list is made by looking at the "CLIMSOFT Issues" database. The document describes the problems found in Climsoft, some of which are already solved in the new release. Here only the issues that are still unresolved or delayed to version 4 are listed:

- Delay to version 4:

- **Question 5:** Station Details and Location
  When attempting to locate a station through the "Search Station" box, I am able to see a list of names, which is good, but when I select one, none of the related information (station ID, station name, country etc.) is displayed. The left and right arrows on either side of the record box do a good job scrolling through the stations. Could we get the "Search Station" box to do the same?

- **Question 7:** Possible 2 part Installation. I don't know how the setup file is created (InstallShield perhaps?) but I'm wondering whether it might be a good idea to separate the databases from the setup file. This might help in the long term and with Roger's idea of a 2-stage installation. So users would install the software, then they would unzip the databases. It might then be easier to clear the databases of any test data. My suspicion is that currently the tables in the database are populated to some extent during testing but are not cleared before the setup file is re-created. We need to differentiate between records in tables that form part of the CLIMSOFT system (e.g. the core element list) and records that are test data or data relevant to particular sites. I will try and clarify this in the Data Model document for which I could really do with having a copy of the source files. When it comes to the instrument list there are currently 7 records in the table but only 4 of these have anything beyond a code. When you set a Station Element the form only shows the instrument name and consequently there appear to be blanks for 3 of the instruments in the list.

- **Question 8:** The data model here is strange as you can specify an element/instrument combination that doesn't make sense –for instance you could specify temperature as being measured by a raingauge which is clearly nonsense.

- **Question 10:** Data for Some Elements for One Day"
  In some cases when looking at the element abbreviations it is not obvious what the elements are. This information can be found if you have access to the metadata, but as I understand it, only some users will have this permission. Is there a way to bring up the table containing all the elements for key entry users?

- **Question 18:** Station Details and Location
  In the database that comes with the latest version of the software there is a station already entered for Machakos (I think I have spelt it correctly). Ideally the database should be blank. When trying to delete this station – which I eventually managed to do – I got messages about not being able to delete it due to related records – fair enough but a bit more information would be useful here explaining what needs to be deleted and in what order – most users would not understand about the relationships in the database and a message about related records might be off-putting for some users. Also even when I had successfully deleted the station record it was still showing in the Station Details and Location dialog.

- **Question 20:** Station Location
  When I click on Update in the Station Location section, it tells me that the operation was successful but then clears the record. If I go to view I can see the data I just entered but it isn't displaying in the form. This dialog/form is particularly confusing with the Details

section and the Location section seeming to work independently from each other. Ideally they should be linked and I think it would be easier to understand if there was a main form for the station details which included a sub-form for the station location. That way users could clearly see the link between station and location and that a station can have several locations.

- **Question 23:** Station Details and Location
  If I have two or more location records for a station I can only ever see one of them in the Station Details and Location Window – I would have thought you would be able to scroll between the different locations for the same station. All locations show if I choose View in the Location section.

- **Question 37:** Station Details:
  I have discovered some data that I believe was used in CLICOM as tutorial data. In that the sites include a comment field so there is information such as:
  - 200m from buildings, fully exposed
  - Sited on campus of Meteorology Department of Limmoccow's main University, 35m from nearest building 10m high. Sunshine recorder on top of building.
  - Visibility, wind speed using Beaufort scale, wind direction and sunshine from roof of hospital. This has a view of flat open ground and small and large trees.
  - I'm wondering if a similar comments field would be useful in CLIMSOFT for describing aspects of the station location or is this what the Features section is for? If so how do you use the Features section as I notice this is greyed out and therefore not accessible in the Metadata dialog?

- **Question 38:** Drop-down lists
  Would it be possible for all drop-down lists to allow automatic completion in the same way that Combo boxes in Access do? For example if I wanted to enter the station name into the following dialog:
  I could start typing and as I type the "K" it would display the first station in the list that started with K. Currently I either have to select from the list or type the value in full.

- **Question 39:** In this dialog box the field is labelled "Station name or identifier" but the drop-down list only shows the station name and as there is no requirement for the station name to be unique this can be confusing. For example the CLICOM data I found has some stations with the same names but different IDs, so the drop-down list appears to show repeats as you can see below. Perhaps the station name should be unique if it is being used in this list without the ID.

- **Question 41:** Complete Element List
  When updating the elements in the metadata section is there a simpler way of making the selection of elements required? Working through each element in the form is tedious with over 600 elements and the view list although it shows all elements, the list is too long and is not in an easy to use order. My suggestions are to:
  - Drastically reduce the number of standard elements;
  - Already reduced to the commonly used ones
  - the list according to the element name rather than the code and have this sort within a

sort on “selected” so that all the currently selected elements appear first followed by the non-selected elements.

- Instead of displaying 0 or -1 in the total and selected columns can we have a check-box displayed as I’m sure this would make it clearer – most non-database users would not know that 0 was “no” and -1 was “yes”.
- Do we really need both “element_name “ and “description” – they seem to be much the same in most cases.
- Likewise do we really need both “units” and “element_type” – these are the same for most cases.
- It would be good if we could move around the list using the <tab> key rather than always having to click in the cell you want to move to.
- It would also be useful to be able to scroll the list using the mouse wheel

- **Question 50:** Producing Products Guide
  The contents of this guide, as well as this part of the program, are not yet decided and more information regarding the products and how to create them is needed.
- **Question 52:** Products – Data
  The list of elements displayed does not correspond to the type of element selected. We need to make only minute data available when we select that option, hourly when hourly is selected and daily when daily is selected
- **Question 54:** Products – Backup
  This should be moved to the DB utilities section
- **Question 56:** This is a comment from SASCAL
  Problem: Unable to produce an inventory of a bigger database (many stations) the system hangs.
  Improvement to be made: Command prompt to be used as its effective to do that job.
- **Question 58:** This is a comment from SASCAL
  Problem: Not compatible with all Operating Systems (eg Linux, windows)
  Improvement to be made: Should be scalable
- **Question 60:** This is a comment from SASCAL
  Problem: Not easy to use when transferring data from older version to users (needs a programmer)
  Improvement to be made: Any upgrade produced should be simpler to the user for them to do the job better.
- **Question 79:** This is a comment from SASCAL
  Problem: Unable to upload data into the databases especially when the network is slow(times out)
  Improvement to be made: Should be easier to dump data into the database using the MySQL commands (other than ODBC only)
- **Question 80:** This is a comment from SASCAL
  Problem: Unable to show the detected error when uploading data into MYSQL main

(always users repeat the same procedure)
Improvement to be made: Should indicate the value or where the uploading stops (e.g. if it's a value at 10/10/2011) not users doing trail & error

- **Question 81:** This is a comment from SASCAL
  Problem: Unable to transfer data between different databases (e.g. MySQL main old to MySQL main new)
  Improvement to be made: It should be easier to manipulate the data in whatever format and database might be.

- Unresolved:
  - **Question 6:** (Related to question 3) An issue that links to the data model is the fact that currently you can have several stations with the same name. For example, in my example data I have 3 stations called "Fantorr" – the IDs are all different but of course the drop-down list only shows the station names, therefore it is impossible to see which one is the one you want. I would suggest that "station name" is made unique in the data model so this problem doesn't arise.
  - **Question 13:** On the form for "Data for some elements for one day (shown below) how difficult would it be to display the element description or name rather than the abbreviation? For example rather than "TMPMIN" it could display "Temperature: Minimum".
  - **Question 24:** Station Aliases
    The ID Alias section is confusing. Looking at what's in the database there are 5 alias types: rainfall_id, local_id, wmo_id, iata_id and icao_id. I think I worked out that icao_id is the same as "aviationid" although I can't remember how I worked that out and it's not obvious from the names used. The ID alias button takes you to the screen capture included. Ideally it should only show the aliases for the current station not for all stations. When you click the ID Alias button you would have a particular station record displayed and it would be useful if the list was automatically populated with this station ID.
  - **Question 26:** Updating the Intermediate Database
    Outside of the "selected forms" option I have not checked whether the options function correctly as I do not have the necessary data (Clicom Daily CSV, CLIMSOFT v2 Access main db etc). Could I get data that would allow me to test these options?
  - **Question 28:** AWS: Creating a New Structure
    In the existing AWS guide there is no mention of the "Element Details" Window. In the existing records I have, it seems that it is used interchangeably with "Element Name" or in some cases contains the same information.
  - **Question 29:** AWS: Processing
    In the existing manual it describes a "Structure" box; however, I don't see this box in my version of CLIMSOFT, I see "Structure Name". Is this the same thing? Also in the existing manual it says "if data is in a different format then "OTHER" should be selected. I do not see this option unless it is in the "Delimiter" line, in which case it should be moved.

- **Question 34:** Producing Products – Data
  In the existing guide it says "Data products are produced either from the daily or hourly observational records" however, when opening the data window, there are a number of other choices available. Should these be disabled?
- **Question 40:** I can't get this import to work. I have a file that is now formatted with the columns going up to "31" followed by "FLAG-1" through to FLAG-31". I go to Key Entry to start (I wonder if there should be an import button directly on the Welcome screen as it seems counter-intuitive to use "Key Entry" when I'm wanting to import). I get to the …Single key-entry mode… window, so double-click on Data for one element for the whole month – my data file has each row as one month of data for a particular element. I get the "Data for one element for the Whole month" form.
- **Question 54:** Double Key Entry
  I select double key entry mode and choose the temp work file as the file to be used as a reference which works fine. I then select "data for one element for the whole month" and begin re-entering data. However, when re-entering data it seems that when the program is checking the current values against the ones in the temp work file it is selecting the data in the first row of the hour column instead of the data in the first row of the day one field to compare.
- **Question 55:** Importing from MYSQL Main v2 Database
  I do not have a MYSQL V2 backup file so have not verified that data transfers smoothly.
- **Question 56:** Encoding and Transmitting Data though FTP
  I do not have an ftp server so have not verified that data can be sent via this method.
- **Question 62:** MetaData – Importing Schedule Data
  When Schedule Data are imported from a .csv file some of the existing cases are overwritten.
- **Question 63:** MetaData – Schedule records not importing correctly
  The number of schedule records in the database is not updating correctly on the Updating Temporary Database dialog. To try and work out what was happening here I removed all existing records from the Schedule table – I checked to make sure they were gone from all 3 databases. I then imported my 6 Schedule records from my .csv file and I ended up with just 4 records in the file. As you can see from the dialog it is adding 4 and updating 2 because there are 2 records for midnight and midday and it seems to be overwriting these instead of bringing them all in.
- **Question 66:** Updating the intermediate database
  I am still having trouble getting the data that is entered into a temporary workfile to move into the intermediate database. I have to create a new station in order for CLIMSOFT to refresh and transfer the data. Is there a way to create a refresh button which would allow one to update changes that have been made in the temporary workfile?

- **Question 68:** AWS Guide: Element settings
  The instructions in this portion of the guide have not been verified. Could they be checked by someone with an understanding of BUFR and how it interacts with CLIMSOFT
- **Question 69:** Data Entry Guide:
  Samuel, could you read Cathy's description of "period cells" and confirm it is correct.
- **Question 73:** Producing Products: Plotting Charts
  The information in the guide for plotting charts is incomplete. If this product is functional it will need a fuller explanation.
- **Question 82:** When trying to customize a form and selecting "data for some elements for one day" I receive the following error, "Run Time Error 3265. Item not found in this collection".
  The program then closes.
- ***Question 83:*** *Metadata:*
  In the manual it is stressed the importance of following an specifi order in the insertion of metadata. However, the way it is written is not too clear. It would be better to show the steps clearly:
  1) Station; 2) element; 3) Instrument; 4) Station element; 5) Schedule; 6) Schedule Class
- ***Question 84:*** *Obs_element table:*
  The "obs_element" table should be available in different languages (one possibility would be to have one table for each language, and one table with the identifiers, which should be always the same?).
- ***Question 85:*** *station_location & lookup_location:*
  The data type of the attribute "begin_datetime" from the "lookup_location" table of the temp db is TEXT, whereas the date type of the "begin_datetime" from the "station_location" table of the intermediate db is DATE/TIME
  In general, make sure that all "dates" attributes are described as "DATE/TIME".
- ***Question 86:*** *obs_element:*
  I could not find the element "Days_of_precipitation" in Climsoft v3.2
- ***Question 88:*** *Documentation of QC:*
  Instructions about the "Quality Controlcould be included also in the User's Guide, not only in the Administrator's guide.
  Documentation about the QC method used in the QC tool would be essential, including a description of the Flag system.
- ***Question 89:*** *QC:*
  After correcting "wrong" data that was identified by the QC from Climsoft, corrected values were given to substitute the wrong ones. They are saved in the ".csv" file but when trying to update the data from the db it does not work. An example of this:
  Absolute limits check:
  1) The "absolute limit check" is applied, getting a "csv" file with the "wrong" values.
  2) The wrong values are changed in the ".csv" file and after that the file is saved with the

same name (it would be better to keep the "wrong" values for later checks)
3) After clicking on "Update" and selecting the changed ".csv" file (in this example "qc_absolute_limits_check_output.csv"), Climsoft says that the update was successful.
4) When checking the "observation" table from the "intermediate_db", it was observed that the values remain the same (the "wrong" values are not replaced).

- ***Question 90:*** *QC:*
  There is a problem in the "qc_report", since it does not always have an hour field ("hh"). However, some of the queries linked to this report ask for the hour file and, when it is not found, the query does not work anymore. This is the case of "qry_qc_excel_output_abs_limits_check".
  The "hh" field appears in "qc_report" when it is linked to the ".csv" file qc_data_products.csv", but not when it is linked to the "qc_absolute_limits_check_output.csv".

Other tables in the database

So far this document has concentrated on the tables in the database that form part of the data model – i.e. those that are related to each other in some way. In addition there are 41 tables that are not in the data model. Many of these are created from queries in the database and are effectively storage areas for outputs from CLIMSOFT.

The list of additional tables is shown below in Table 2~~Table 5~~.

Table 2~~5~~ - Tables not in the Data Model

Table name	Comment
Climat_all_groups	
Climat_coded_missing_days	
Climat_coded_missing_years	Identical structure to table above
Climat_drybulb_thresholds	
Climat_elements_for_mean	Includes 4 records with element codes 2, 3, 4 and 5
Climat_extreme_highest	
Climat_extreme_lowest	
Climat_header	
Climat_initialized_coded_missing_years	12 records with element codes 2, 3, 5, 18, 84, 93, 101, 106, 107, 110, 111 and 166 – **coded_missing** = "30" for all records
Climat_initialized_missing_days	One value – **missing_days = "31"**
Climat_initialized_missing_years	Same as table **climat_initialized_coded_missing_years** except that the **coded_missing** field is numeric
Climat_missing_days	**Elemet_code** is text (and is spelt wrong!)
Climat_missing_years	

Climat_quintiles_ordered_years	**Year_order** = 1 to 30
Climat_thirty_year_monthly_rainfall_quintiles	
Climat_thirty_year_monthly_rainfalls_ordered_by_month	
Climdex_daily_data	
Data_availability	
Data_inventory	22 records for station "JWANENG AIRPORT" – tables should be cleared out.
Data_products	
Elements_for_inventory	
Fao_daily_data_selected_stations	
Flags	Includes codes M=Missing, T=Trace, E=Estimated, G=Generated
Generation_of_daily_from_hourly	
Hour_list	00:00 through to 23:00
Instat_daily_data	
Inter_element_comparisons_output	
Julian_day	Lists day of year for leap years and non-leap years
Longterm_monthly_drybulb_std_mean	
Longterm_monthly_means	
Longterm_monthly_rainday_means	
Month_days	Lists number of days in each month
Month_list	Lists the months numerically and by abbreviation – and sets March as 1
Pentad_classes	Seems to divide month into sets of 5 day periods
Quintiles	Seems to have 30 records per month and 5 quintiles per month with 6 records per quintile?
Seq_day	Seems to be day numbers 1 to 31
Seq_month	Month numbers 1 to 12
Seq_year	Only one record which is 2006
Stations_for_inventory	Has one station ID which is 9137089 – looks like it wasn't cleared out
Tdcf_bufr_climat	No idea what this is but there are 179 records
Thirty_year_monthly_rainfall_totals	

*I suspect many of these tables are remnants either from previous versions of CLIMSOFT or from other installations. Are all these tables needed? I can imagine some such as **Julian_day**, **Month_list** and some others are used within the CLIMSOFT code but I'm not convinced that they are all needed and pruning them would make a more understandable data structure. It would be good to document these tables explaining*

what role they play within the system. I feel this would be a useful exercise both for version 3.# and as we move towards development of version 4.

Queries

There seem to be over 700 queries in the main database.

Are these all actually needed? Many of the queries don't run for a variety of reasons and I suspect that over time some of the table names and/or field names have changed but the queries have never been cleared out. Consequently the database is cluttered with lots of things that are no longer used.

The queries are listed below.

Table 36 -Queries in the CLIMSOFT database

Query Name	Query Type
Copy Of qry_inventory_step3_data_gaps_append	Append
Copy Of qry_restore_data_from_backup	Append
qry_append_climat_header	Append
qry_append_climat_stn_id	Append
qry_append_daily_mean_from_hourly	Append
qry_append_daily_value_from_hourly	Append
qry_append_datetime_from_backup	Append
qry_append_flags_from_intermediate_backup	Append
qry_append_missing_code_to-element	Append
qry_append_RH_to_obs_table	Append
qry_append_RH_to_observation_table	Append
qry_append_section1_groups	Append
qry_append_TD_to_observation_table	Append
qry_append_values_from_intermediate_backup	Append
qry_aws_append_daily_precip_to_observation	Append
qry_aws_append_datetime_for_daily	Append
qry_climat_append_actual_missing_days	Append
qry_climat_append_actual_missing_years	Append
qry_climat_append_initialized_coded_missing_years	Append
qry_climat_append_initialized_missing_days	Append
qry_climat_append_initialized_missing_element_days	Append
qry_climat_append_initialized_missing_years	Append
qry_climat_append_initialized_years	Append
qry_climat_append_list_of_coded_missing_days	Append
qry_climat_append_longterm_drybulb	Append
qry_climat_append_longterm_drybulb_std_dev	Append
qry_climat_append_section4_group0	Append
qry_climat_append_section4_group1	Append
qry_climat_append_section4_group2	Append
qry_climat_append_section4_group3	Append
qry_climat_append_section4_group4	Append
qry_climat_append_section4_group5	Append
qry_climat_append_thirty_year_rainfall_quintiles	Append
qry_climat_extreme_highest	Append
qry_climat_extreme_lowest	Append
qry_climat_extreme_ttt_dailymean_highest	Append

Query Name	Query Type
qry_climat_extreme_ttt_dailymean_lowest	Append
qry_climat_get_longterm_monthly_means	Append
qry_climat_get_longterm_monthly_stdev_mean	Append
qry_climat_get_mean_monthly_rainydays	Append
qry_climat_get_station_monthly_rainfall_quintiles	Append
qry_climat_missing_years	Append
qry_climat_section1_group1	Append
qry_climat_section1_group2	Append
qry_climat_section1_group3	Append
qry_climat_section1_group4	Append
qry_climat_section1_group5	Append
qry_climat_section1_group6	Append
qry_climat_section1_group7	Append
qry_climat_section1_group8	Append
qry_climat_section1_group9	Append
qry_climat_section2_group0	Append
qry_climat_section2_group1	Append
qry_climat_section2_group2	Append
qry_climat_section2_group3	Append
qry_climat_section2_group4	Append
qry_climat_section2_group5	Append
qry_climat_section2_group6	Append
qry_climat_section2_group7	Append
qry_climat_section2_group8	Append
qry_climat_section2_group9	Append
qry_climat_section3_group0	Append
qry_climat_section3_group1	Append
qry_climat_section3_group2	Append
qry_climat_section3_group3	Append
qry_climat_section3_group4	Append
qry_climat_section3_group5	Append
qry_climat_section3_group6	Append
qry_climat_section3_group7	Append
qry_climat_section3_group8	Append
qry_climat_section3_group9	Append
qry_import_from_backup	Append
qry_inventory_stats	Append
qry_longterm_mean_monthly_rainydays_for_climat	Append
qry_longterm_monthly_mean_rainfall_for_climat	Append
qry_longterm_monthly_mean_sunshine_for_climat	Append
qry_longterm_monthly_mean_tmax_for_climat	Append
qry_longterm_monthly_mean_tmean_for_climat	Append
qry_longterm_monthly_mean_tmin_for_climat	Append
qry_longterm_monthly_mean_TT_StDev_for_climat	Append
qry_process_climat	Append
qry_qc_values_exceeding_local_monthly_limits	Append
qry_stn_elem_stats	Append
qry_update_datetime_from_intermediate	Append
qry_update_element_from_intermediate	Append
qry_update_level_from_intermediate	Append
qry_update_location_from_intermediate	Append

Query Name	Query Type
qry_update_metadata_element	Append
qry_update_metadata_instrument	Append
qry_update_metadata_scheduleclass	Append
qry_update_metadata_station	Append
qry_update_metadata_stationelement	Append
qry_update_observation_from_another_main	Append
qry_update_observation_from_intermediate	Append
qry_update_station_from_intermediate	Append
Copy Of qry_annual_total_by_month_step1	Crosstab
qry_annual_mean_by_month_step1	Crosstab
qry_annual_total_by_month_order_step1	Crosstab
qry_annual_total_by_month_step1	Crosstab
qry_climat_arrange_coded_missing_days	Crosstab
qry_climat_quintiles_missing_data_check	Crosstab
qry_climat_section1_groups_transposed	Crosstab
qry_climat_section2_missing_years	Crosstab
qry_daily_data_cross_tab_initial	Crosstab
qry_daily_data_crosstab	Crosstab
qry_daily_data_elements_crosstab	Crosstab
qry_daily_data_for_instat	Crosstab
qry_data_products	Crosstab
qry_DayByMonth	Crosstab
qry_get_dekadal_for_agrometshell_crosstab	Crosstab
qry_monthly_dailymean_crosstab	Crosstab
qry_monthly_dailymean_season_month_cross_tab	Crosstab
qry_monthly_total_crosstab	Crosstab
qry_monthly_total_season_month_cross_tab	Crosstab
qry_select_oldest_stns	Crosstab
qry_windrose_input_data	Crosstab
qry-climat_section2_missing_years	Crosstab
qry_elms_inventory	Data Definition
qry_stns_inventory	Data Definition
Copy Of qry_daily_data_output	Make Table
Copy Of qry_dekadal_total_agrometshell_output	Make Table
Copy Of qry_extreme_highest_daily_value_by_date	Make Table
Copy Of qry_extreme_lowest_daily_value_by_date	Make Table
qry_annual_dailymean_jan_dec_crosstab_report	Make Table
qry_annual_dailymean_jul_jun_crosstab_report	Make Table
qry_annual_rainfall_total_jan_dec_crosstab_report	Make Table
qry_annual_rainfall_total_jul_jun_crosstab_report	Make Table
qry_annual_total_by_month_order_step2	Make Table
qry_annual_total_by_month_step2_backup	Make Table
qry_aws_30min	Make Table
qry_daily_data_output	Make Table
qry_dekadal_total_agrometshell_output	Make Table
qry_dekadal_total_for_agrometshell_output	Make Table
qry_dewpoint_from_drybulb_and_wetbulb	Make Table
qry_element1_element2_synop_db_comparison	Make Table
qry_element2_for_qc_synop_db	Make Table
qry_extreme_highest_dek_daily_mean_output	Make Table
qry_extreme_highest_dek_total	Make Table

Query Name	Query Type
qry_hourly_data_output	Make Table
qry_instat_daily_data	Make Table
qry_inventory_step3	Make Table
qry_inventory_step3_backup	Make Table
qry_inventory_step3_data_gaps_append	Make Table
qry_longterm_monthly_daily_mean_output	Make Table
qry_wind_dir_sectors_for_windrose	Make Table
qry_wind_ff_for_windrose	Make Table
Query1	Make Table
basic_daily_rain	Select
climat_0 08 022_threshold_days_with_precip_gt_150	Select
climat_0 08 022_threshold_days_with_precip_gt_5	Select
climat_0 08 022_threshold_days_with_precip_gt_50	Select
Copy of Copyqry_climat_0 08 022_threshold_days_snowdepth_gt_50cm	Select
Copy of Coqry_climat_0 08 022_threshold_days_snowdepth_gt_10cm	Select
Copy Of qry_annual_total_by_month_step2	Select
Copy of qry_climat_0 08 022_threshold_days_snowdepth_gt_1cm	Select
Copy of qry_climat_bufr_0 08 023_first_order_stats_cancel	Select
Copy of qry_climat_bufr_0 08 052_code_days_ff_gt_20	Select
Copy of qry_climat_bufr_0 08 052_code_days_ff_gt_20m_per_sec	Select
Copy of qry_climat_bufr_0 08 052_code_days_ff_gt_30	Select
Copy of qry_climat_bufr_0 08 052_code_days_vis_lt_50m	Select
Copy Of qry_daily_data	Select
Copy Of qry_daily_data_for_annual_total_step1	Select
Copy Of qry_daily_data_for_rainy_days	Select
Copy Of qry_daily_flags	Select
Copy Of qry_highest_daily_value	Select
Copy Of qry_hourly_data	Select
Copy Of qry_inventory_step3_for_data_gaps	Select
Copy Of qry_lowest_daily_value	Select
Copyqry_climat_bufr_0 04 022_short_time_period_section2	Select
qr_inventory_step2_backup	Select
qrry_climat_bufr_0 08 020_missing_years_tmax_tmin	Select
qry__converted_f_to_c	Select
qry_annual_average	Select
qry_annual_daily_mean	Select
qry_annual_dailymean_jan_dec_crosstab	Select
qry_annual_dailymean_jul_jun_crosstab	Select
qry_annual_mean_by_month_step2	Select
qry_annual_rainfall_total_jan_dec_crosstab	Select
qry_annual_rainfall_total_jul_jun_crosstab	Select
qry_annual_total	Select
qry_annual_total_by_month_step2	Select
qry_annual_total_report	Select
qry_available_obs_dates	Select
qry_aws_converted_hourly_to_daily	Select
qry_aws_daily_precip_from_hourly	Select
qry_aws_hourly_data_precip_time_booked_back	Select
qry_climat_bufr_0 04 022_first_order_stats__mean_section2	Select
qry_climat_bufr_0 04 022_time_period_section2	Select
qry_climat_bufr_0 07 004_monthly_stn_pressure	Select

Query Name	Query Type
qry_climat_bufr_0 07 032_first_order_stats_cancel_section2	Select
qry_climat_bufr_0 08 020_missing_years_tmax_tmin_step1	Select
qry_climat_bufr_0 12 101_section2_monthly_tmean	Select
qry_climat_bufr_0 12 151_monthly_sunshine	Select
qry_climat_bufr_002_0 01 001_wmo_block_number	Select
qry_climat_bufr_003_0 01 002_wmo_station_number	Select
qry_climat_bufr_004_0 01 015_station_name	Select
qry_climat_bufr_005_0 02 001_station_type	Select
qry_climat_bufr_006_0 04 001_year	Select
qry_climat_bufr_007_0 04 002_month	Select
qry_climat_bufr_008_0 04 001_day	Select
qry_climat_bufr_009_0 04 004_hour	Select
qry_climat_bufr_010_0 04 005_minute	Select
qry_climat_bufr_011_0 05 001_latitude	Select
qry_climat_bufr_012_0 06 001_longitude	Select
qry_climat_bufr_013_0 07 030_elevation	Select
qry_climat_bufr_014_0 07 031_height_barometer	Select
qry_climat_bufr_015_0 04 074_short_time_displacement	Select
qry_climat_bufr_016_0 04 023_days_in_month	Select
qry_climat_bufr_017_0 08 023_first_order_stats_mean_value	Select
qry_climat_bufr_018_0 10 004_section_2_stn_pressure	Select
qry_climat_bufr_019_0 10 051_pppp_msl_pressure	Select
qry_climat_bufr_020_0 07 004_pressure_standard_level	Select
qry_climat_bufr_021_0 10 009_geopotential	Select
qry_climat_bufr_022_0 07 032_height_of_sensor_above_ground_tt	Select
qry_climat_bufr_023_0 12 101_drybulb	Select
qry_climat_bufr_024_0 02 051_sensor_indicator_tt	Select
qry_climat_bufr_025_0 04 051_time_of_obs_tmax	Select
qry_climat_bufr_026_0 12 118_tmax	Select
qry_climat_bufr_027_0 04 052_time_of_obs_tmin	Select
qry_climat_bufr_028_0 12 119_tmin	Select
qry_climat_bufr_029_0 13 004_eee_vapour_pressure	Select
qry_climat_bufr_030_0 08 023_first_order_stats_cancel	Select
qry_climat_bufr_031_0 12 151_drybulb_stdev	Select
qry_climat_bufr_032_0 07 032_sensor_height_cancel	Select
qry_climat_bufr_033_0 08 050_missing_days_code_pressure	Select
qry_climat_bufr_034_0 08 020_misssing_days_stn_level_pressure	Select
qry_climat_bufr_035_0 08 050_missing_days_code_drybulb	Select
qry_climat_bufr_036_0 08 020_missing_days_drybulb_tmean	Select
qry_climat_bufr_037_0 08 050_missing_days_code_vapour_pressure	Select
qry_climat_bufr_038_0 08 020_missing_days_eee_vapour_pressure	Select
qry_climat_bufr_039_0 08 050_missing_days_code_tmax	Select
qry_climat_bufr_040_0 08 020_missing_days_tmax	Select
qry_climat_bufr_041_0 08 050_missing_days_code_tmin	Select
qry_climat_bufr_042_0 08 020_missing_days_tmin	Select
qry_climat_bufr_043_0 14 032_sss_sunshine	Select
qry_climat_bufr_044_0 14 033_sss_sunshine_percentage	Select
qry_climat_bufr_045_0 08 050_missing_days_code_sunshine	Select
qry_climat_bufr_046_0 08 020_missing_days_sss_sunshine	Select
qry_climat_bufr_047_0 08 052_code_days_ff_gt_10m_per_sec	Select
qry_climat_bufr_048_0 08 022_threshold_days_gt_f10f10	Select

Query Name	Query Type
qry_climat_bufr_049_0 08 052_code_days_ff_gt_20	Select
qry_climat_bufr_050_0 08 022_threshold_days_gt_f20f20	Select
qry_climat_bufr_051_0 08 052_code_days_ff_gt_30	Select
qry_climat_bufr_052_0 08 022_threshold_days_gt_f30f30	Select
qry_climat_bufr_053_0 08 052_code_days_tmax_lt_0	Select
qry_climat_bufr_054_0 08 022_threshold_days_lt_tx0tx0	Select
qry_climat_bufr_055_0 08 052_code_days_tmax_gt_25	Select
qry_climat_bufr_056_0 08 022_threshold_days_gt_t25t25	Select
qry_climat_bufr_057_0 08 052_code_days_tmax_gt_30	Select
qry_climat_bufr_058_0 08 022_threshold_days_gt_t30t30	Select
qry_climat_bufr_059_0 08 052_code_days_tmax_gt_35	Select
qry_climat_bufr_060_0 08 022_threshold_days_gt_t35t35	Select
qry_climat_bufr_061_0 08 052_code_days_tmax_gt_40	Select
qry_climat_bufr_062_0 08 022_threshold_days_gt_t40t40	Select
qry_climat_bufr_063_0 08 052_code_days_tmin_lt_0	Select
qry_climat_bufr_064_0 08 022_threshold_days_lt_tn0tn0	Select
qry_climat_bufr_065_0 08 052_code_days_snowdepth_gt_0	Select
qry_climat_bufr_066_0 08 022_threshold_days_snowdepth_gt_0	Select
qry_climat_bufr_067_0 08 052_code_days_snowdepth_gt_1cm	Select
qry_climat_bufr_068_0 08 022_threshold_days_snowdepth_gt_1cm	Select
qry_climat_bufr_069_0 08 052_code_days_snowdepth_gt_10cm	Select
qry_climat_bufr_070_0 08 022_threshold_days_code_snow_gt_10cm	Select
qry_climat_bufr_071_0 08 052_code_days_snowdepth_gt_50cm	Select
qry_climat_bufr_072_0 08 022_threshold_days_code_snow_gt_50cm	Select
qry_climat_bufr_073_075_0 08 052_code_days_vis_lt_50m	Select
qry_climat_bufr_074_0 08 022_threshold_days_vis_lt_50m	Select
qry_climat_bufr_075_0 08 052_code_days_vis_lt_100m	Select
qry_climat_bufr_076_0 08 022_threshold_days_vis_lt_100m	Select
qry_climat_bufr_077_0 08 052_code_days_vis_lt_1000m	Select
qry_climat_bufr_078_0 08 022_threshold_days_vis_lt_1000m	Select
qry_climat_bufr_079_0 08 052_code_days_with_hail	Select
qry_climat_bufr_080_0 08 022_threshold_days_with_hail	Select
qry_climat_bufr_081_0 08 052_code_days_with_thunder	Select
qry_climat_bufr_082_0 08 022_threshold_days_with_thunder	Select
qry_climat_bufr_083_0 07 032_height_of_sensor_above_groung_tt	Select
qry_climat_bufr_084_0 08 053_day_qualifier_highest_tmean	Select
qry_climat_bufr_085_0 04 003_day_highest_tmean	Select
qry_climat_bufr_086_0 12 152_value_highest_tmean	Select
qry_climat_bufr_087_0 08 053_day_qualifier_lowest_tmean	Select
qry_climat_bufr_088_0 04 003_day_lowest_tmean	Select
qry_climat_bufr_089_0 12 153_value_lowest_tmean	Select
qry_climat_bufr_090_0 08 053_day_qualifier_highest_tmax	Select
qry_climat_bufr_091_0 04 003_day_highest_tmax	Select
qry_climat_bufr_092_0 08 023_first_order_stats_max_value	Select
qry_climat_bufr_093_0 12 101_value_highest_tmean	Select
qry_climat_bufr_094_0 08 053_day_qualifier_lowest_tmin	Select
qry_climat_bufr_095_0 04 003_day_lowest_tmin	Select
qry_climat_bufr_096_0 08 023_first_order_stats_minimum_value	Select
qry_climat_bufr_097_0 12 101_value_lowest_tmin	Select
qry_climat_bufr_098_0 08 023_first_order_stats_cancel	Select
qry_climat_bufr_099_0 07 032_sensor_height_above_ground_wind	Select

Query Name	Query Type
qry_climat_bufr_100_0 07 032_instrument_type_wind	Select
qry_climat_bufr_101_0 08 053_day_qualifier_highest_wind_speed	Select
qry_climat_bufr_102_0 04 003_day_highest_wind_speed	Select
qry_climat_bufr_103_0 11 046_value_highest_wind_speed	Select
qry_climat_bufr_104_0 08 053_day_qualifier_cancel	Select
qry_climat_bufr_105_0 04 003_day_precip_data	Select
qry_climat_bufr_106_0 04 004_hour_precip_data	Select
qry_climat_bufr_107_0 04 023_days_in_month_precip	Select
qry_climat_bufr_108_0 07 032_height_of_sensor_precip	Select
qry_climat_bufr_109_0 13 060_precip	Select
qry_climat_bufr_110_0 13 051_monthly_rainfall_quintile	Select
qry_climat_bufr_111_0 04 053_days_precip_ge_1	Select
qry_climat_bufr_112_0 08 050_missing_days_code_precip	Select
qry_climat_bufr_113_0 08 020_missing_days_precip	Select
qry_climat_bufr_114_0 08 052_code_days_precip_gt_1mm	Select
qry_climat_bufr_115_0 08 022_threshold_days_with_precip_gt_1	Select
qry_climat_bufr_116_0 08 052_code_days_precip_gt_5mm	Select
qry_climat_bufr_117_0 08 022_threshold_days_with_precip_gt_5	Select
qry_climat_bufr_118_0 08 052_code_days_precip_gt_10mm	Select
qry_climat_bufr_119_0 08 022_threshold_days_with_precip_gt_10	Select
qry_climat_bufr_120_0 08 052_code_days_precip_gt_50mm	Select
qry_climat_bufr_121_0 08 022_threshold_days_with_precip_gt_50	Select
qry_climat_bufr_122_0 08 052_code_days_precip_gt_100mm	Select
qry_climat_bufr_123_0 08 022_threshold_days_with_precip_gt_100	Select
qry_climat_bufr_124_0 08 052_code_days_precip_gt_150mm	Select
qry_climat_bufr_125_0 08 022_threshold_days_with_precip_gt_150	Select
qry_climat_bufr_126_0 08 053_day_qualifier_highest_precip	Select
qry_climat_bufr_127_0 04 003_day_highest_precip	Select
qry_climat_bufr_128_0 13 052_value_highest_precip	Select
qry_climat_bufr_129_0 07 032_sensor_height_precip_cancel	Select
qry_climat_bufr_130_0 04 001_reference_year_start	Select
qry_climat_bufr_131_0 04 001_reference_year_end	Select
qry_climat_bufr_132_0 04 002_month_section2	Select
qry_climat_bufr_133_0 04 003_day_section2	Select
qry_climat_bufr_134_0 04 004_hour_section2	Select
qry_climat_bufr_135_0 04 074_short_time_displacement_section2	Select
qry_climat_bufr_136_0 04 022_short_time_period_section2	Select
qry_climat_bufr_137_0 08 023_first_order_stats_mean_section2	Select
qry_climat_bufr_138_0 10 004_p0p0p0p0_stn_level_pressure	Select
qry_climat_bufr_139_0 10 051_section2_monthly_msl_pressure	Select
qry_climat_bufr_140_0 07 004_pressure_std_level	Select
qry_climat_bufr_141_0 10 009_section2_monthly_geopotential	Select
qry_climat_bufr_142_10 07 032_sensor_height_tt_section2	Select
qry_climat_bufr_143_0 12 101_value_highest_tmax	Select
qry_climat_bufr_144_0 02 051_sensor_indicator_tt_section2	Select
qry_climat_bufr_145_0 04 052_obs_time_tmax	Select
qry_climat_bufr_146_0 12 118_section2_monthly_tmax	Select
qry_climat_bufr_147_0 04 052_obs_time_tmin	Select
qry_climat_bufr_148_0 12 119_section2_monthly_tmin	Select
qry_climat_bufr_149_0 13 004_section2_monthly_vapour_pressure	Select
qry_climat_bufr_150_0 12 151_section2_monthly_tmean_stdev	Select

Query Name	Query Type
qry_climat_bufr_151_0 07 032_sensor_height_cancel	Select
qry_climat_bufr_152_0 14 032_section2_monthly_sunshine	Select
qry_climat_bufr_153_0 08 023_first_order_stats_cancel	Select
qry_climat_bufr_154_0 04 001_reference_year_end_precip_section2	Select
qry_climat_bufr_155_0 04 001_reference_year_start_precip	Select
qry_climat_bufr_156_0 04 002_month__precip_section2	Select
qry_climat_bufr_157_0 04 003_day_precip_section2	Select
qry_climat_bufr_158_0 04 004_hour_precip_section2	Select
qry_climat_bufr_159_0 04 022_time_period_precip_section2	Select
qry_climat_bufr_160_0 07 032_height_of_sensor_precip_section2	Select
qry_climat_bufr_161_0 08 023_first_order_stats_mean_precip	Select
qry_climat_bufr_162_0 13 060_section2_monthly_total_precip	Select
qry_climat_bufr_163_0 04 053_section2_nrnr_mean_rainy_days	Select
qry_climat_bufr_164_0 08 023_first_order_stats_cancel_section2	Select
qry_climat_bufr_165_0 08 050_missing_years_code_pressure	Select
qry_climat_bufr_166_0 08 020_missing_years_stn_pressure	Select
qry_climat_bufr_167_0 08 050_missing_years_code_tmean	Select
qry_climat_bufr_168_0 08 020_missing_years_tmean	Select
qry_climat_bufr_169_0 08 050_missing_years_code_tmax_tmin	Select
qry_climat_bufr_170_0 08 020_missing_years_tmax_tmin	Select
qry_climat_bufr_171_0 08 050_missing_years_code_vapour_pressure	Select
qry_climat_bufr_172_0 08 020_missing_years_vapour_pressure	Select
qry_climat_bufr_173_0 08 050_missing_years_code_precip	Select
qry_climat_bufr_174_0 08 020_missing_years_precip	Select
qry_climat_bufr_175_0 08 050_missing_years_code_sunshine	Select
qry_climat_bufr_176_0 08 020_missing_years_sunshine	Select
qry_climat_bufr_177_0 08 050_missing_years_code_tmax	Select
qry_climat_bufr_178_0 08 020_missing_years_tmax	Select
qry_climat_bufr_179_0 08 050_missing_years_code_tmin	Select
qry_climat_bufr_180_0 08 020_missing_years_tmin	Select
qry_climat_bufr_station_identification	Select
qry_climat_coded_count_precip_ge_1	Select
qry_climat_coded_missing_days_all_elements	Select
qry_climat_coded_monthly_rainfall_total	Select
qry_climat_coded_temp	Select
qry_climat_coded_tmax_and_sign	Select
qry_climat_coded_tmin_and_sign	Select
qry_climat_count_days_with_hail	Select
qry_climat_count_days_with_thunder	Select
qry_climat_count_of_rainy_days	Select
qry_climat_count_precip_ge_1	Select
qry_climat_count_precip_gt_10	Select
qry_climat_count_precip_gt_100	Select
qry_climat_count_precip_gt_150	Select
qry_climat_count_precip_gt_5	Select
qry_climat_count_precip_gt_50	Select
qry_climat_count_snowdepth_gt_0	Select
qry_climat_count_snowdepth_gt_1	Select
qry_climat_count_snowdepth_gt_10	Select
qry_climat_count_snowdepth_gt_50	Select
qry_climat_count_tmax_ge_25	Select

Query Name	Query Type
qry_climat_count_tmax_ge_30	Select
qry_climat_count_tmax_ge_35	Select
qry_climat_count_tmax_ge_40	Select
qry_climat_count_tmax_lt_0	Select
qry_climat_count_tmin_lt_0	Select
qry_climat_count_visibility_lt_1000m	Select
qry_climat_count_visibility_lt_100m	Select
qry_climat_count_visibility_lt_50m	Select
qry_climat_count_windspeed_gt_10kt	Select
qry_climat_count_windspeed_gt_20kt	Select
qry_climat_count_windspeed_gt_30kt	Select
qry_climat_daily_data	Select
qry_climat_daily_data1	Select
qry_climat_days_with_hail	Select
qry_climat_days_with_precip_ge_1	Select
qry_climat_days_with_precip_ge_10	Select
qry_climat_days_with_precip_ge_100	Select
qry_climat_days_with_precip_ge_150	Select
qry_climat_days_with_precip_ge_5	Select
qry_climat_days_with_precip_ge_50	Select
qry_climat_days_with_snowdepth_gt_0	Select
qry_climat_days_with_snowdepth_gt_1	Select
qry_climat_days_with_snowdepth_gt_10	Select
qry_climat_days_with_snowdepth_gt_50	Select
qry_climat_days_with_thunder	Select
qry_climat_days_with_tmax_ge_25	Select
qry_climat_days_with_tmax_ge_30	Select
qry_climat_days_with_tmax_ge_35	Select
qry_climat_days_with_tmax_ge_40	Select
qry_climat_days_with_tmax_lt_0	Select
qry_climat_days_with_tmin_lt_0	Select
qry_climat_days_with_visibility_lt_1000m	Select
qry_climat_days_with_visibility_lt_100m	Select
qry_climat_days_with_visibility_lt_50m	Select
qry_climat_days_with_windspeed_gt_10kt	Select
qry_climat_days_with_windspeed_gt_20kt	Select
qry_climat_days_with_windspeed_gt_30kt	Select
qry_climat_drybulb_and_missing	Select
qry_climat_eee_and_missing	Select
qry_climat_elements_section1_groups_8_9	Select
qry_climat_extreme_drybulb_dailymean_highest	Select
qry_climat_extreme_drybulb_dailymean_lowest	Select
qry_climat_geopotential_and_missing	Select
qry_climat_get_drybulb_longterm_std_dev	Select
qry_climat_get_longterm_drybulb_avg	Select
qry_climat_get_month	Select
qry_climat_get_rainydays_by_month	Select
qry_climat_get_selected_month	Select
qry_climat_header	Select
qry_climat_header_and_month	Select
qry_climat_missing_data	Select

Query Name	Query Type
qry_climat_monthly_data	Select
qry_climat_monthly_extremes_highest	Select
qry_climat_monthly_extremes_highest_precip	Select
qry_climat_monthly_extremes_highest_tmax	Select
qry_climat_monthly_extremes_highest_tmean	Select
qry_climat_monthly_extremes_highest_windspeed	Select
qry_climat_monthly_extremes_lowest	Select
qry_climat_monthly_extremes_lowest_tmean	Select
qry_climat_monthly_extremes_lowest_tmin	Select
qry_climat_monthly_initialized_missing_code	Select
qry_climat_monthly_initialized_missing_years	Select
qry_climat_monthly_missing_mean	Select
qry_climat_monthly_rainfall_and_missing	Select
qry_climat_monthly_rainfall_quintile_penultimate	Select
qry_climat_monthly_rainfall_quintiles_initial	Select
qry_climat_msl_pressure_and_missing	Select
qry_climat_rainy_days	Select
qry_climat_rainy_days_day_range	Select
qry_climat_section1_group_6	Select
qry_climat_section1_groups	Select
qry_climat_section1_groups_and_indicators	Select
qry_climat_section2_coded_drybulb	Select
qry_climat_section2_coded_drybulb_stdev	Select
qry_climat_section2_coded_eee	Select
qry_climat_section2_coded_rain	Select
qry_climat_section2_coded_std_level_press	Select
qry_climat_section2_coded_stnpress	Select
qry_climat_section2_coded_sunshine	Select
qry_climat_section2_coded_tmax	Select
qry_climat_section2_coded_tmin	Select
qry_climat_section2_groups_8_9	Select
qry_climat_section2_mean_rainydays	Select
qry_climat_section3_group0_t25t25	Select
qry_climat_section3_group0_t30t30	Select
qry_climat_section3_group1_t35t35	Select
qry_climat_section3_group1_t40t40	Select
qry_climat_section3_group2_tn0tn0	Select
qry_climat_section3_group2_tx0tx0	Select
qry_climat_section3_group3_r01r01	Select
qry_climat_section3_group3_r05r05	Select
qry_climat_section3_group4_r10r10	Select
qry_climat_section3_group4_r50r50	Select
qry_climat_section3_group5_r100r100	Select
qry_climat_section3_group5_r150r150	Select
qry_climat_section3_group6_s00s00	Select
qry_climat_section3_group6_s01s01	Select
qry_climat_section3_group7_s10s10	Select
qry_climat_section3_group7_s50s50	Select
qry_climat_section3_group8_f10f10	Select
qry_climat_section3_group8_f20f20	Select
qry_climat_section3_group8_f30f30	Select

Query Name	Query Type
qry_climat_section3_group9_v1v1	Select
qry_climat_section3_group9_v2v2	Select
qry_climat_section3_group9_v3v3	Select
qry_climat_section4_highest_precip	Select
qry_climat_section4_highest_t_avg	Select
qry_climat_section4_highest_tmax	Select
qry_climat_section4_highest_tmean	Select
qry_climat_section4_highest_windmax	Select
qry_climat_section4_lowest_tmean	Select
qry_climat_section4_lowest_tmin	Select
qry_climat_stn_id	Select
qry_climat_stn_pressure_and_missing	Select
qry_climat_stn_rainfall_and_missing	Select
qry_climat_stn_yy_mm_section1_group1	Select
qry_climat_stn_yy_mm_section1_group2	Select
qry_climat_stn_yy_mm_section1_group3	Select
qry_climat_stn_yy_mm_section1_group4	Select
qry_climat_stn_yy_mm_section1_group6	Select
qry_climat_stn_yy_mm_section1_group7	Select
qry_climat_sunshine	Select
qry_climat_sunshine_and_missing	Select
qry_climat_temp_mean_missing_and_sign	Select
qry_climat_ttt_daily_mean	Select
qry_climat_wx_hail_for_section4_grp6	Select
qry_climat_wx_number_haildays	Select
qry_climat_wx_number_tsdays	Select
qry_climat_wx_ts_for_section4_grp6	Select
qry_climdex_data_inventory	Select
qry_comparison_between_entered_and_generated_dewpoint	Select
qry_converted_f_to_c	Select
qry_daily_data	Select
qry_daily_data_climdex_output	Select
qry_daily_data_Convert_Tmax_from_F	Select
qry_daily_data_convert_tmin_from_F	Select
qry_daily_data_crosstab_with_total	Select
qry_daily_data_elements_crosstab_julian_day	Select
qry_daily_data_for_annual_mean_step1	Select
qry_daily_data_for_annual_total_step1	Select
qry_daily_data_for_climat_backup	Select
qry_daily_data_for_climat_longterm_means	Select
qry_daily_data_for_instat_output	Select
qry_daily_data_for_inter_stn_comparison	Select
qry_daily_data_for_rainy_days	Select
qry_daily_data_for_selected_period	Select
qry_daily_flags	Select
qry_daily_flags_backup	Select
qry_daily_values_above_threshold	Select
qry_database_backup	Select
qry_database_backup_output	Select
qry_db_backup	Select
qry_dekadal_daily_mean	Select

Query Name	Query Type
qry_dekadal_daily_mean_output	Select
qry_dekadal_for_agrometshell_grouped_by_year	Select
qry_dekadal_mean	Select
qry_dekadal_total	Select
qry_dekadal_total_output	Select
qry_drybulb_wetbulb_comparison_QC	Select
qry_element1_element2_comparison	Select
qry_element1_for_qc	Select
qry_element2_for_qc	Select
qry_extract_dewpoint_for_RH_calculation	Select
qry_extract_drybulb_for_RH_calculation	Select
qry_extract_wetbulb_for_RH_calculation	Select
qry_extreme_highest_annual_total	Select
qry_extreme_highest_daily_value_by_date	Select
qry_extreme_highest_dek_daily_mean	Select
qry_extreme_highest_monthly_daily_count_by_year	Select
qry_extreme_highest_monthly_daily_mean_by_year	Select
qry_extreme_highest_monthly_total	Select
qry_extreme_highest_seasonal_total	Select
qry_extreme_lowest_annual_total	Select
qry_extreme_lowest_daily_value_by_date	Select
qry_extreme_lowest_dek_daily_mean	Select
qry_extreme_lowest_dek_total	Select
qry_extreme_lowest_monthly_daily_count_by_year	Select
qry_extreme_lowest_monthly_daily_mean_by_year	Select
qry_extreme_lowest_monthly_total	Select
qry_extreme_lowest_seasonal_total	Select
qry_extreme_max_daily_value	Select
qry_extreme_min_daily_value	Select
qry_generate_dewpoint_from_drybulb_and_wetbulb	Select
qry_generate_RH_from_TT_and_Td	Select
qry_generation_of_daily_from_hourly	Select
qry_get_integer_for_element_scale_factor	Select
qry_get_inventory_beginyear_and_endyear	Select
qry_get_month_dek_for_agrometshell	Select
qry_get_station_id_integer	Select
qry_highest_annual_total	Select
qry_highest_daily_value	Select
qry_highest_daily_value_in_a_month	Select
qry_highest_daily_value_in_a_month_by_date	Select
qry_highest_daily_value_in_a_year	Select
qry_highest_daily_value_in_a_year_by_date	Select
qry_highest_dek_mean	Select
qry_highest_dekadal_total	Select
qry_highest_monthly_daily_count	Select
qry_highest_monthly_daily_mean	Select
qry_highest_monthly_total	Select
qry_highest_seasonal_total	Select
qry_hourly_data	Select
qry_hourly_data_backup	Select
qry_hourly_data_for_daily_mean	Select

Query Name	Query Type
qry_hourly_data_for_daily_value	Select
qry_import_daily	Select
qry_import_flags_from_intermediate_db	Select
qry_import_flags_from_intermediate_db_backup	Select
qry_import_from_intermediate_datetime	Select
qry_import_from_intermediate_scale	Select
qry_import_from_intermediate_station_id	Select
qry_import_oldest_stations	Select
qry_import_synop	Select
qry_import_values_from_intermediate_backup	Select
qry_import_values_from_intermediate_db	Select
qry_import_values_from_intermediate_db_backup	Select
qry_inter_station_comparison	Select
qry_inter_stn_comparison_with_abs_diff	Select
qry_inventory	Select
qry_inventory_daily_data_step1	Select
qry_inventory_data_gaps_begindate_enddate	Select
qry_inventory_output	Select
qry_inventory_step1	Select
qry_inventory_step1_for_data_gaps	Select
qry_inventory_step2	Select
qry_inventory_step3_for_data_gaps	Select
qry_longterm_average_monthly_total	Select
qry_longterm_mean_monthly_count	Select
qry_longterm_mean_monthly_count_output	Select
qry_longterm_monthly_daily_mean	Select
qry_longterm_monthly_mean_synoptic_data	Select
qry_lookup_element	Select
qry_lookup_element_daily	Select
qry_lookup_element_hourly	Select
qry_lookup_hour	Select
qry_lookup_station	Select
qry_lowest_annual_total	Select
qry_lowest_daily_value	Select
qry_lowest_daily_value_in_a_month	Select
qry_lowest_daily_value_in_a_month_by_date	Select
qry_lowest_dek_mean	Select
qry_lowest_dekadal_total	Select
qry_lowest_monthly_daily_count	Select
qry_lowest_monthly_daily_mean	Select
qry_lowest_monthly_total	Select
qry_lowest_seasonal_total	Select
qry_make_same_date_width	Select
qry_modify_monthly_limits	Select
qry_monthly_daily_count	Select
qry_monthly_daily_count_output	Select
qry_monthly_daily_mean	Select
qry_monthly_daily_mean_output	Select
qry_monthly_dailymean_season_month	Select
qry_monthly_mean_from_daily	Select
qry_monthly_mean_synoptic_data	Select

Query Name	Query Type
qry_monthly_rainydays_for_climat	Select
qry_monthly_summary_for_climat_stage1	Select
qry_monthly_summary_for_climat_stage2	Select
qry_monthly_total	Select
qry_monthly_total_for_quintiles	Select
qry_monthly_total_from_daily	Select
qry_monthly_total_output	Select
qry_monthly_total_season_month	Select
qry_monthy_mean_from_daily_output	Select
qry_number_of_days_in_month	Select
qry_oldest_rainfall_stns	Select
qry_pentad_rainfall	Select
qry_qc_element1	Select
qry_qc_element2	Select
qry_qc_monthly_limits	Select
qry_qc_values_exceeding_local_monthly_limits_backup	Select
qry_restore_data_from_backup	Select
qry_saturation_vapour_press_at_drybulb	Select
qry_saturation_vapour_press_at_wetbulb	Select
qry_seasonal_mean	Select
qry_seasonal_summary_by_month_july_to_june	Select
qry_seasonal_total	Select
qry_select_data_for_upload_to_main	Select
qry_select_data_for_upload_to_main_step1	Select
qry_select_dekads_for_agrometshell	Select
qry_select_stations_before_1900	Select
qry_seq_year_month_day	Select
qry_station_list	Select
qry_stations_climdex_workshop_brazzaville_200704	Select
qry_stations_for_wis_xml_bhutan	Select
qry_stations_for_wis_xml_rwanda	Select
qry_stations_for_wis_xml_sudan	Select
qry_stations_for_wis_xml_zimbabwe	Select
qry_stn_elem_available_begin_date_and_end_date	Select
qry_stn_elem_available_obs_days	Select
qry_stn_elem_expected_obs_days	Select
qry_stn_elem_inventory	Select
qry_stn_loc	Select
qry_synoptic_data	Select
qry_tmax_tmin_validation	Select
qry_view_climat_missing_data_for_rr_quintiles	Select
qry_windrose_direction_and_speed_final	Select
qry_windrose_direction_and_speed_initial	Select
qry_windrose_hourly_direction	Select
qry_windrose_hourly_speed	Select
qry_wmo_id	Select
Copy Of qry_instat_update_feb29_for_non_leap_year_to_9988	Update
qry_clear_month_order_in_table_month_list	Update
qry_climat_bufr_clear_climat_values_in_tdcf_table	Update
qry_instat_code_trace_to_-9999	Update
qry_instat_update_feb29_for_non_leap_year_to_9988	Update

Query Name	Query Type
qry_instat_update_trace_to_-9999	Update
qry_mark_data_availability_backup	Update
qry_mark_obs_records_as_updated	Update
qry_update_date	Update
qry_update_metadata	Update
qry_update_station_monthly_limits	Update

Annex 1 – Database Diagram of Climsoft v4

Draft 1

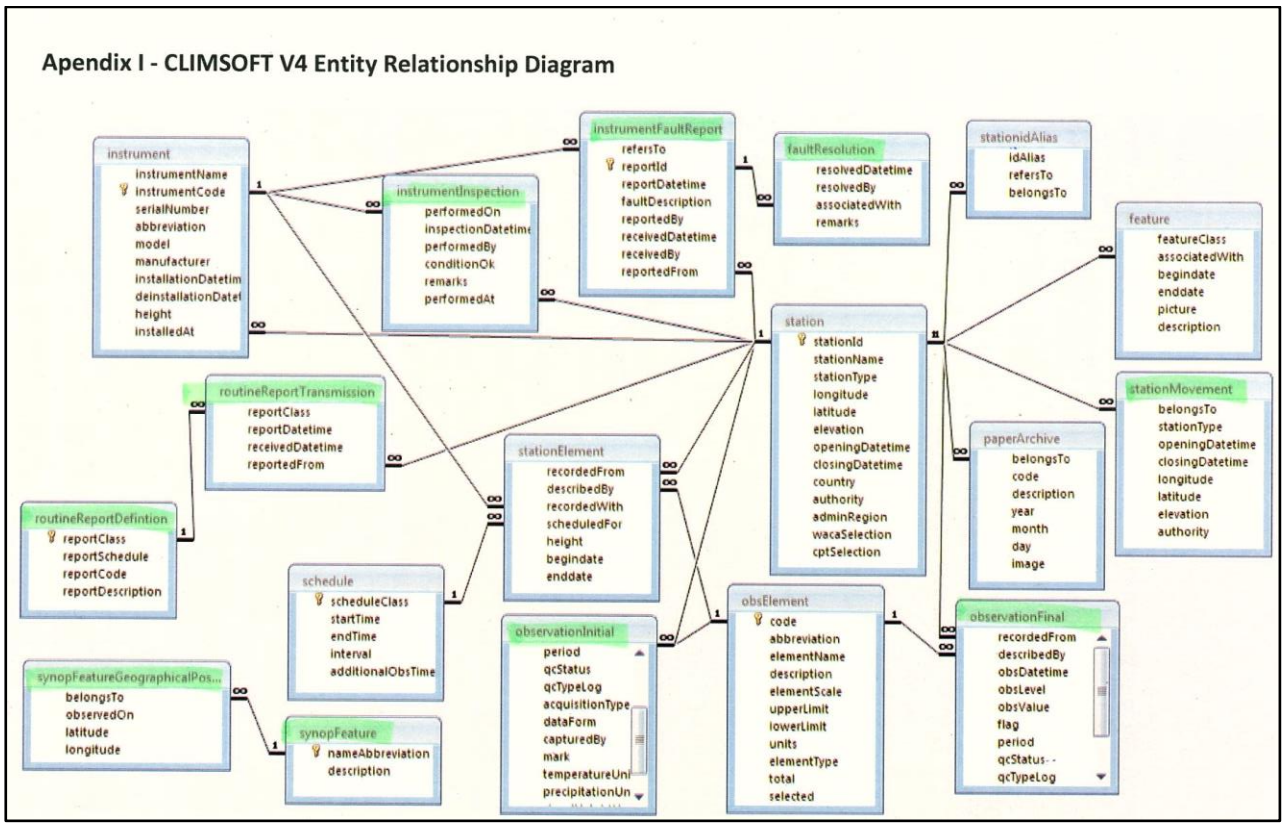


Figure 5. The new E/R Diagram proposed by Albert Mhanda for Climsoft v4 during the Climsoft Meeting in Kenya (July 2014).

Draft 2

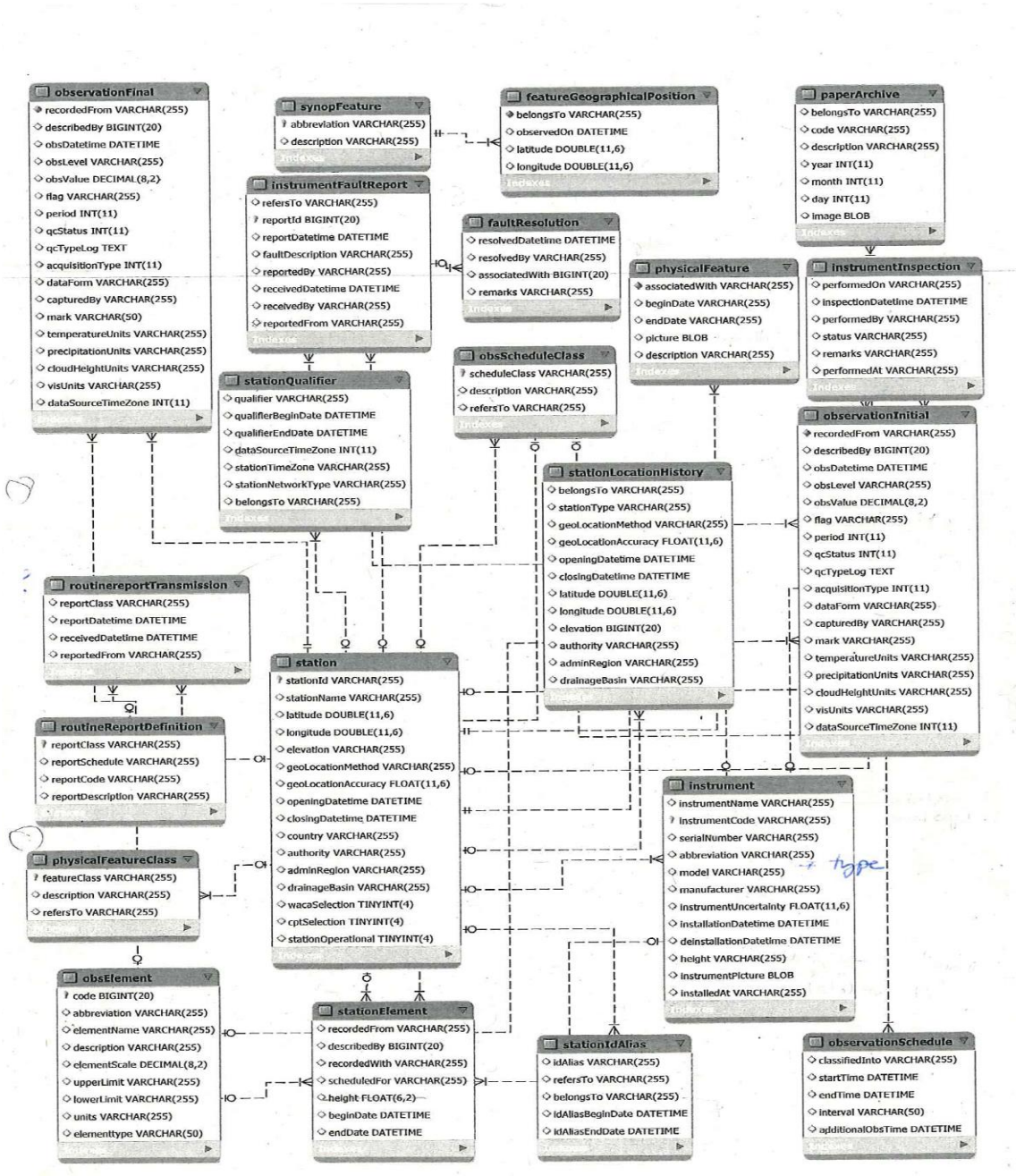


Figure 6. The new E/R Diagram proposed by Albert Mhanda for Climsoft v4 during the Climsoft Meeting in MetOffice (December 2014).