FDA XML Data Format Design Specification

Barry Brown, Product Integration Manager, Mortara Instrument
Mark Kohls, Engineering Director, GE Medical Systems-Information Technologies
Norman Stockbridge, M.D., Ph. D., Medical Team Leader, Food and Drug Administration
Others Welcome…..
Send us your information….

1 Introduction

1.1 Purpose

This specification document exists for the purpose of defining the design of the FDA XML Data Format (FDADF) and ensuring that all interested individuals and organizations involved with the project have the same understanding of the data format design.

1.2 Scope of Specification

This document covers the design for the waveform data format as well as relevant submission information. Areas addressed by this document include identifying design elements that meet the requirements for the data set previously specified in the FDA XML Data Format Requirements Specification[1] which was initiated after the FDA's meeting on November 19th, 2001[2]. Applicability and interaction with other standards bodies and data definitions, e.g. CDISC and HL-7, as well as current practice with SAS submission data sets is also discussed. Design for the structure of the data on electronic media is also covered.

2 Overview

2.1 Background

New Drug Application (NDA) sponsors collect biological data, often as waveforms from subjects dosed with the candidate drug. A number of measurements are made from the data, or from close derivations of that data. Those measurements are compiled into datasets and statistically analyzed. The datasets are submitted with the NDA to support the findings.

2.1.1 Issue

The FDA would like to get a sense of the accuracy and consistency of the measurements made from the collected biological data. The FDA cannot do this without being given the opportunity to view the biological data used for making those measurements.

2.1.2 Goal

To facilitate the submission of the biological data or close derivations of it used to make the measurements. The biological data should be annotated with points and intervals to show the reviewer relevant landmarks used for making the measurements.

2.1.3 Process Description

During the course of a drug *study*, a *subject* is given a dosage of some compound, either the drug under study or a placebo. Periodically, *recording devices* collect biological data from the subject. Each *recording session* is typically made up of one or more periodically sampled channels *(waveforms).* Sometimes measurements are made on the raw waveforms themselves, and in other times on transformations of that data into other domains. Therefore, the data from which measurements are made can be in many different forms: electrical-potential vs. sample-time, electrical-potential vs. cycle-time, pressure vs. sample-time, power vs. frequency, or possible future requirements. No matter what domain the data from which measurements are made is in, the data is related to a period of real-time, the period of time from which the recording was made.

2.2 Recording

Typically a recording device will periodically sample one or more biological sensors. The sample values will generally represent biological parameters, e.g. temperature, pressure, oxygen saturation, electrical potential, at each time point. Customarily the samples are plotted vs. time and the resulting waveform imparts meaning to a clinician. Additional information can be derived from the waveforms by making measurements on the waveforms themselves; these measurements are usually included in the statistical datasets provided to the FDA.

However, the "sensor-value vs. time" waveform is not the only set of data measurements can be made from. For example, if the waveform is cyclical in nature (e.g. ECG, BP), an average cycle can be derived. Measurements can be made on that average cycle and can be used for further statistical analysis. Other derivations can be imagined, for example, analyzing the frequency content of the average cycle. Viewing power versus frequency might give an idea of how much energy is in a certain part of the frequency spectrum, and this may be useful for certain types of analysis.

3 Data Format

3.1 Technology

The FDA desires to use XML as the underlying technology for the specification. This matches the Agency's strategic direction for data submissions. It is also aligned with other industry initiatives such as the CDISC's Submission Data Model (SDM).

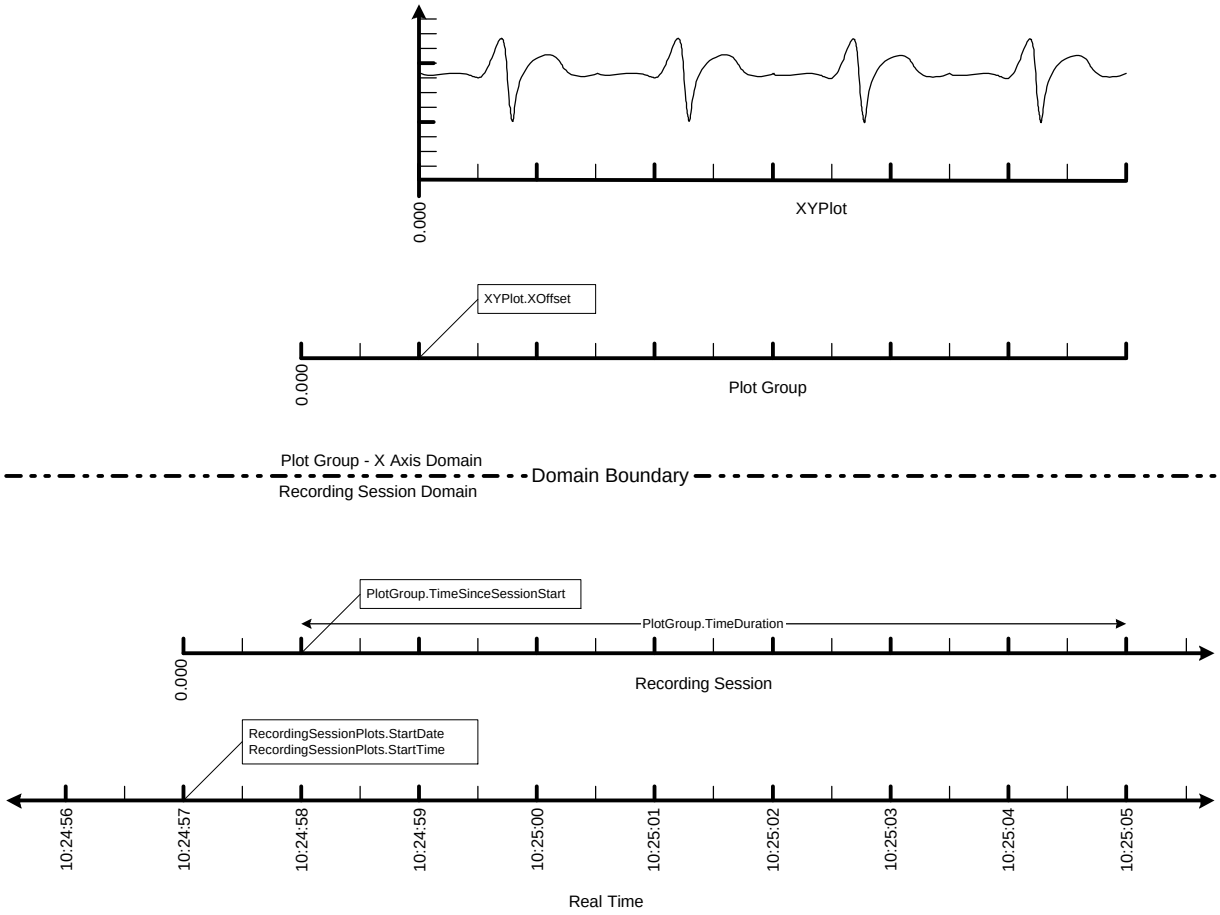
3.2 Description

The data is assumed to be two-dimensional (2-D) in nature. It is also assumed that the data or the data from which it is derived was collected from a subject in a drug study. The *recording session* producing the source data has a real start time and duration. Datasets can, therefore, be related to real-time, even if real-time is not one of the dimensions.

The 2-D datasets can also be annotated. The annotations will give the FDA reviewer domain-specific landmarks demonstrating how the data was used. The annotations are intended to give a precise indication of where measurements and fiducial points, e.g. R-peak, QRS-onset, and T-offset, used in analysis were made or placed. Some computed measurements, e.g. QTc, cannot be directly included in the waveform annotation, but would be supplied in the corresponding submission data.

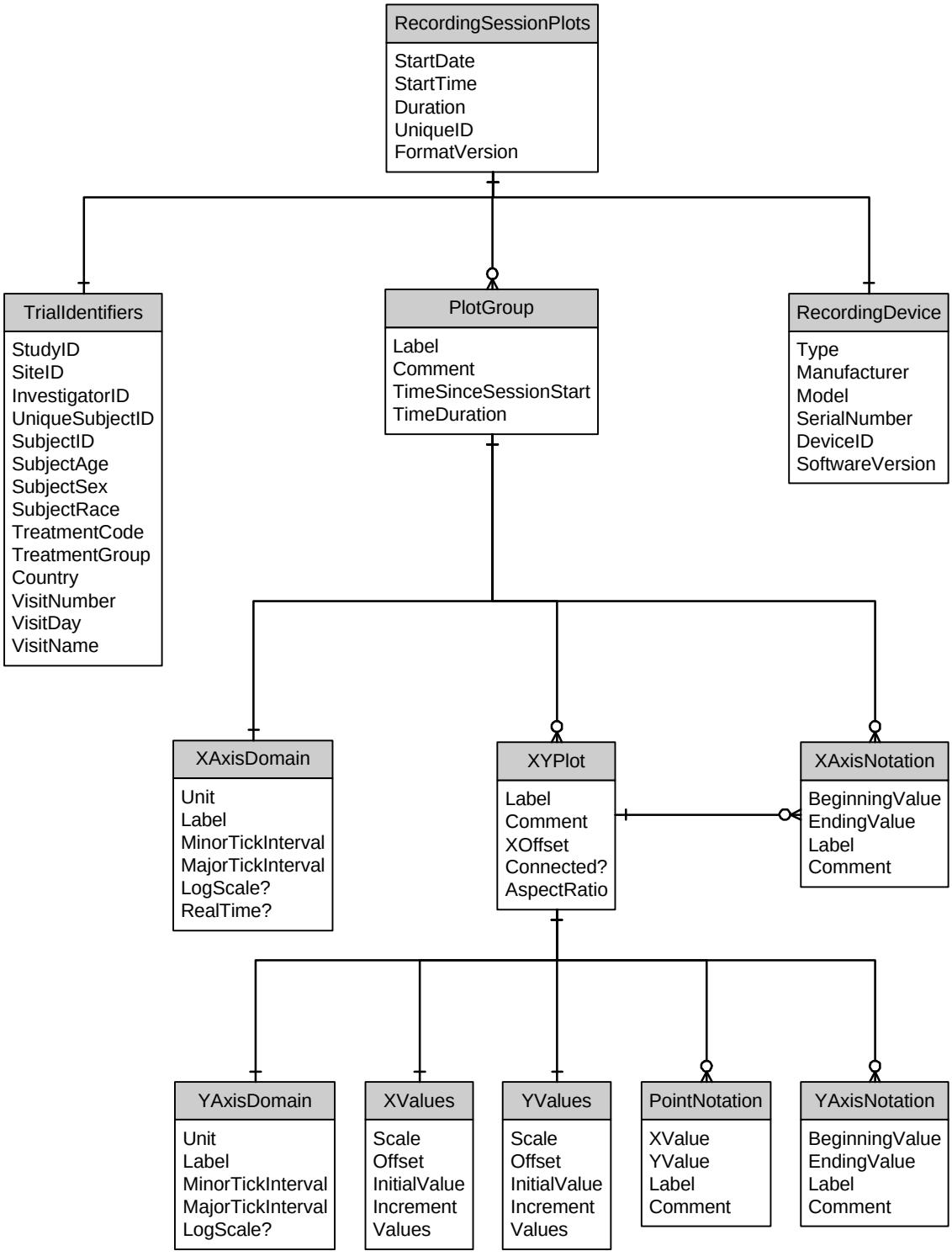
The 2-D datasets can be grouped together when they are related by a common X-axis dimension. For example, ECG rhythm leads can be grouped together because they share a common X-axis "sample-time" dimension. Leads of an ECG median beat are related by a common "cycle-time" X-axis dimension. The group of datasets can therefore share common annotations made in the X-axis domain.

The relationships between real-time, the recording session, dataset groups and datasets can be thought of as a set of related coordinate systems. Dataset groups can optionally share a dimension that is related to the recording session time-domain by simple translate/scale transformations. If the group is not directly related, e.g. the common dimension of a median beat is cycle-time, not real-time, the group can only be related to a period of recording session time, but cannot be directly plotted along the recording session timeline. Therefore, each dataset group will have attributes relating it to the period of recording session time it is derived from, but will not describe the relationship as a pure translate/scale transformation.


0.000
XYPlot
XYPlot.XOffset
0.000
Plot Group
Plot Group - X Axis Domain
Domain Boundary
Recording Session Domain
PlotGroup.TimeSinceSessionStart
PlotGroup.TimeDuration
0.000
Recording Session
RecordingSessionPlots.StartDate
RecordingSessionPlots.StartTime
10:24:56
10:24:57
10:24:58
10:24:59
10:25:00
10:25:01
10:25:02
10:25:03
10:25:04
10:25:05
Real Time

3.3 The Entity-Relationship Model

The following diagram shows the entities and their relationships. The ”crow’s foot“ connector shows a “many to one” relationship. For example, “RecordingSessionPlots” may contain 0 or more “PlotGroups”, and a “PlotGroup” must contain exactly one “XAxisDomain”.



3.4 XML DTD

Below is the suggested DTD for a RecordingSessionPlots:

```
<!ELEMENT RecordingSessionPlots (StartDate, StartTime, Duration?, UniqueID?, TrialIdentifiers?,
RecordingDevice?, PlotGroup*)>
    <!ELEMENT StartDate (#PCDATA)>
    <!ELEMENT StartTime (#PCDATA)>
    <!ELEMENT Duration  (#PCDATA)>
    <!ELEMENT UniqueID  (#PCDATA)>
<!ATTLIST RecordingSessionPlots FormatVersion CDATA 1.0>

<!ELEMENT TrialIdentifiers (StudyID?, SiteID?, InvestigatorID?, UniqueSubjectID?, SubjectID?,
SubjectAge?, SubjectAgeUnits?, SubjectSex?, SubjectRace?, TreatmentCode?, TreatmentGroup?,
Country?, VisitNumber?, VisitDay?, VisitName?)>
    <!ELEMENT StudyID         (#PCDATA)>
    <!ELEMENT SiteID          (#PCDATA)>
    <!ELEMENT InvestigatorID  (#PCDATA)>
    <!ELEMENT SubjectID       (#PCDATA)>
    <!ELEMENT SubjectAge      (#PCDATA)>
    <!ELEMENT SubjectSex      (#PCDATA)>
    <!ELEMENT SubjectRace     (#PCDATA)>
    <!ELEMENT TreatmentCode   (#PCDATA)>
    <!ELEMENT TreatmentGroup  (#PCDATA)>
    <!ELEMENT Country         (#PCDATA)>
    <!ELEMENT VisitNumber     (#PCDATA)>
    <!ELEMENT VisitDay        (#PCDATA)>
    <!ELEMENT VisitName       (#PCDATA)>

<!ELEMENT RecordingDevice (Type?, Manufacturer?, Model?, SerialNumber?, DeviceID?,
SoftwareVersion?)>
    <!ELEMENT Type            (#PCDATA)>
    <!ELEMENT Manufacturer    (#PCDATA)>
    <!ELEMENT Model           (#PCDATA)>
    <!ELEMENT SerialNumber    (#PCDATA)>
    <!ELEMENT DeviceID        (#PCDATA)>
    <!ELEMENT SoftwareVersion (#PCDATA)>

<!ELEMENT PlotGroup (Label, Comment?, TimeSinceSessionStart?, TimeDuration?, XAxisDomain,
XYPlot*, XAxisNotation*)>
    <!ELEMENT Label                (#PCDATA)>
    <!ELEMENT Comment              (#PCDATA)>
    <!ELEMENT TimeSinceSessionStart (#PCDATA)>
    <!ELEMENT TimeDuration         (#PCDATA)>

<!ELEMENT XAxisDomain (Unit, Label, MinorTickInterval?, MajorTickInterval?, LogScale?,
RealTime?)>
    <!ELEMENT Unit             (#PCDATA)>
<!--<!ELEMENT Label            (#PCDATA)> -->
    <!ELEMENT MinorTickInterval (#PCDATA)>
    <!ELEMENT MajorTickInterval (#PCDATA)>
    <!ELEMENT LogScale         (#PCDATA)>
    <!ELEMENT RealTime         (#PCDATA)>

<!ELEMENT XYPlot (Label, Comment?, XOffset?, Connected?, AspectRatio?, YAxisDomain, XValues,
YValues, PointNotation*, YAxisNotation*)>
<!--<!ELEMENT Label       (#PCDATA)> -->
<!--<!ELEMENT Comment     (#PCDATA)> -->
    <!ELEMENT XOffset     (#PCDATA)>
    <!ELEMENT Connected   (#PCDATA)>
    <!ELEMENT AspectRatio (#PCDATA)>

<!ELEMENT XAxisNotation (BeginningValue, EndingValue?, Label, Comment?)>
    <!ELEMENT BeginningValue (#PCDATA)>
    <!ELEMENT EndingValue    (#PCDATA)>
<!--<!ELEMENT Label          (#PCDATA)> -->
<!--<!ELEMENT Comment        (#PCDATA)> -->

<!ELEMENT YAxisDomain (Unit, Label, MinorTickInterval?, MajorTickInterval?, LogScale?)>
<!--<!ELEMENT Unit             (#PCDATA)> -->
<!--<!ELEMENT Label            (#PCDATA)> -->
<!--<!ELEMENT MinorTickInterval (#PCDATA)> -->
```

```
<!--<!ELEMENT MajorTickInterval (#PCDATA)> -->
<!--<!ELEMENT LogScale          (#PCDATA)> -->

<!ELEMENT XValues (Scale?, Offset?, InitialValue?, Increment?, Values?)>
    <!ELEMENT Scale        (#PCDATA)>
    <!ELEMENT Offset       (#PCDATA)>
    <!ELEMENT InitialValue (#PCDATA)>
    <!ELEMENT Increment    (#PCDATA)>
    <!ELEMENT Values       (#PCDATA)>

<!ELEMENT YValues (Scale?, Offset?, InitialValue?, Increment?, Values?)>
<!--<!ELEMENT Scale        (#PCDATA)> -->
<!--<!ELEMENT Offset       (#PCDATA)> -->
<!--<!ELEMENT InitialValue (#PCDATA)> -->
<!--<!ELEMENT Increment    (#PCDATA)> -->
<!--<!ELEMENT Values       (#PCDATA)> -->

<!ELEMENT PointNotation (XValue, YValue, Label, Comment?)>
       <!ELEMENT XValue  (#PCDATA)>
       <!ELEMENT YValue  (#PCDATA)>
<!--<!ELEMENT Label   (#PCDATA)> -->
<!--<!ELEMENT Comment (#PCDATA)> -->

<!ELEMENT YAxisNotation (BeginningValue, EndingValue?, Label, Comment?)>
<!--<!ELEMENT BeginningValue (#PCDATA)> -->
<!--<!ELEMENT EndingValue    (#PCDATA)> -->
<!--<!ELEMENT Label          (#PCDATA)> -->
<!--<!ELEMENT Comment        (#PCDATA)> -->
```

3.5 XML Schema

Below is the suggested XML Schema for a RecordingSessionPlots:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <xs:element name="StartDate"             type="xs:date"/>
  <xs:element name="StartTime"             type="xs:time"/>
  <xs:element name="Duration"              type="xs:duration"/>
  <xs:element name=”UniqueID”              type=”xs:string”/>

  <xs:element name="StudyID"               type="xs:string"/>
  <xs:element name="SiteID"                type="xs:string"/>
  <xs:element name="InvestigatorID"        type="xs:string"/>
  <xs:element name="UniqueSubjectID"       type="xs:string"/>
  <xs:element name="SubjectID"             type="xs:string"/>
  <xs:element name="SubjectAge"            type="xs:duration"/>
  <xs:element name="SubjectRace"           type="xs:string"/>
  <xs:element name="TreatmentCode"         type="xs:string"/>
  <xs:element name="TreatmentGroup"        type="xs:string"/>
  <xs:element name="Country"               type="xs:string"/>
  <xs:element name="VisitNumber"           type="xs:string"/>
  <xs:element name="VisitDay"              type="xs:string"/>
  <xs:element name="VisitName"             type="xs:string"/>

  <xs:element name="DisplayText"           type="xs:string"/>
  <xs:element name="Comment"               type="xs:string"/>
  <xs:element name="TimeSinceSessionStart" type="xs:duration"/>
  <xs:element name="TimeDuration"          type="xs:duration"/>

  <xs:element name=”Type”                  type=”xs:string”/>
  <xs:element name=”Manufacturer”          type=”xs:string”/>
  <xs:element name=”Model”                 type=”xs:string”/>
  <xs:element name=”SerialNumber”          type=”xs:string”/>
  <xs:element name=”DeviceID”              type=”xs:string”/>
  <xs:element name=”SoftwareVersion”       type=”xs:string”/>

  <xs:element name="XOffset"               type="xs:float"/>
  <xs:element name="Connected"             type="xs:boolean"/>
  <xs:element name="AspectRatio"           type="xs:float"/>

  <xs:element name="BeginningValue"        type="xs:float"/>
  <xs:element name="EndingValue"           type="xs:float"/>

  <xs:element name="Unit"                  type="xs:string"/>
  <xs:element name="MinorTickInterval"     type="xs:float"/>
  <xs:element name="MajorTickInterval"     type="xs:float"/>
  <xs:element name="LogScale"              type="xs:boolean"/>
  <xs:element name="RealTime"              type="xs:boolean"/>

  <xs:element name="Scale"                 type="xs:float"/>
  <xs:element name="Offset"                type="xs:float"/>
  <xs:element name="InitialValue"          type="xs:float"/>
  <xs:element name="Increment"             type="xs:float"/>


  <xs:element name="YValue"                type="xs:float"/>
  <xs:element name="XValue"                type="xs:float"/>

  <xs:element name="RecordingSessionPlots">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="StartDate"/>
        <xs:element ref="StartTime"/>
        <xs:element ref="Duration"                                          minOccurs="0"/>
        <xs:element ref=”UniqueID”                                          minOccurs=”0”/>
        <xs:element name="PlotGroup"        type="PlotGroupType”        minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="TrialIdentifiers" type="TrialIdentifiersType" minOccurs="0"/>
        <xs:element name="RecordingDevice"  type="RecordingDeviceType"  minOccurs="0"/>
```

```
    </xs:sequence>
    <xs:attribute name=”FormatVersion” fixed=”1.0”/>
  </xs:complexType>
</xs:element>

<xs:complexType name=”TrialIdentifiersType”>
  <xs:sequence>
    <xs:element ref=”StudyID”/>
    <xs:element ref=”SiteID”          minOccurs=”0”/>
    <xs:element ref=”InvestigatorID”  minOccurs=”0”/>
    <xs:element ref=”UniqueSubjectID” minOccurs=”0”/>
    <xs:element ref=”SubjectID”       minOccurs=”0”/>
    <xs:element ref=”SubjectAge”      minOccurs=”0”/>
    <xs:element name=”SubjectSex” type=”SexType”  minOccurs=”0”/>
    <xs:element ref=”SubjectRace”     minOccurs=”0”/>
    <xs:element ref=”TreatmentCode”   minOccurs=”0”/>
    <xs:element ref=”TreatmentGroup”  minOccurs=”0”/>
    <xs:element ref=”Country”         minOccurs=”0”/>
    <xs:element ref=”VisitNumber”     minOccurs=”0”/>
    <xs:element ref=”VisitDay”        minOccurs=”0”/>
    <xs:element ref=”VisitName”       minOccurs=”0”/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name=”RecordingDeviceType”>
  <xs:sequence>
    <xs:element name=”Type”            minOccurs=”0”/>
    <xs:element name=”Manufacturer”    minOccurs=”0”/>
    <xs:element name=”Model”           minOccurs=”0”/>
    <xs:element name=”SerialNumber”    minOccurs=”0”/>
    <xs:element name=”DeviceID”        minOccurs=”0”/>
    <xs:element name=”SoftwareVersion” minOccurs=”0”/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="PlotGroupType">
  <xs:sequence>
    <xs:element name="Label" type=”PlotGroupLabelType”/>
    <xs:element ref="Comment"                                  minOccurs="0"/>
    <xs:element ref="TimeSinceSessionStart"                    default="P0Y"/>
    <xs:element ref="TimeDuration"                             minOccurs="0"/>
    <xs:element name="XAxisDomain"   type="XAxisDomainType"/>
    <xs:element name="XYPlot"        type="XYPlotType"         minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="XAxisNotation" type="AxisNotationType"  minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="PointNotationType">
  <xs:sequence>
    <xs:element ref="XValue"/>
    <xs:element ref="YValue"/>
    <xs:element name="Label" type=”PointNotationLabelType”/>
    <xs:element ref="Comment" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>


<xs:complexType name="XAxisDomainType">
  <xs:sequence>
    <xs:element ref="Unit"/>
    <xs:element name="Label" type=”XAxisDomainLabelType”/>
    <xs:element ref="MinorTickInterval" minOccurs="0"/>
    <xs:element ref="MajorTickInterval" minOccurs="0"/>
    <xs:element ref="LogScale"          default=”false”/>
    <xs:element ref="RealTime"          default=”false”/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="AxisNotationType">
  <xs:sequence>
    <xs:element ref="BeginningValue"/>
    <xs:element ref="EndingValue" minOccurs="0"/>
```

```
      <xs:element name="Label" type=”AxisNotationLabelType”/>
      <xs:element ref="Comment"     minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="ValuesType">
    <xs:sequence>
      <xs:element ref="Scale"  default=”1”/>
      <xs:element ref="Offset" default=”0”/>
      <xs:choice minOccurs="0">
        <xs:sequence minOccurs="0">
          <xs:element ref="InitialValue" minOccurs="0"/>
          <xs:element ref="Increment"    minOccurs="0"/>
        </xs:sequence>
        <xs:element ref="ValuesList" minOccurs="0"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="XYPlotType">
    <xs:sequence>
      <xs:element name="Label" type=”XYPlotLabelType”/>
      <xs:element ref="Comment"                                    minOccurs="0"/>
      <xs:element ref="XOffset"                                    default=”0”/>
      <xs:element ref="Connected"                                  default=”true”/>
      <xs:element ref="AspectRatio"                                default=”1”/>
      <xs:element name="YAxisDomain"   type="YAxisDomainType"/>
      <xs:element name="XValues"       type="ValuesType"/>
      <xs:element name="YValues"       type="ValuesType"/>
      <xs:element name="PointNotation" type="PointNotationType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="XAxisNotation" type="AxisNotationType"  minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="YAxisNotation" type="AxisNotationType"  minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="YAxisDomainType">
    <xs:sequence>
      <xs:element ref="Unit"/>
      <xs:element name="Label" type=”YAxisDomainLabelType”/>
      <xs:element ref="MinorTickInterval" minOccurs="0"/>
      <xs:element ref="MajorTickInterval" minOccurs="0"/>
      <xs:element ref="LogScale"          default=”false”/>
    </xs:sequence>
  </xs:complexType>

        <xs:simpleType name=”ValuesList”>
               <xs:list itemList=”xs:float”/>
        </xs:simpleType>

  <xs:simpleType name=”SexType”>
    <xs:restriction base=”xs:string”>
      <xs:enumeration value=”M”>
        <xs:annotation><xs:documentation>Male</xs:documentation></xs:annotation>
      </xs:enumeration>
      <xs:enumeration value=”F”>
        <xs:annotation><xs:documentation>Female</xs:documentation></xs:annotation>
      </xs:enumeration>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name=”PlotGroupLabelType”>
    <xs:sequence>
      <xs:element ref=”DisplayText”/>
      <xs:element name=”Code” type=”PlotGroupLabelCodeType” minOccurs=”0”/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name=”PlotGroupLabelCodeType”>
    <xs:restriction base=”xs:string”/>
  </xs:simpleType>
```

```
  <xs:complexType name="AxisNotationLabelType">
    <xs:sequence>
      <xs:element ref="DisplayText"/>
      <xs:element name="Code" type="AxisNotationLabelCodeType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name="AxisNotationLabelCodeType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>

  <xs:complexType name="PointNotationLabelType">
    <xs:sequence>
      <xs:element ref="DisplayText"/>
      <xs:element name="Code" type="PointNotationLabelCodeType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name="PointNotationLabelCodeType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>

  <xs:complexType name="XAxisDomainLabelType">
    <xs:sequence>
      <xs:element ref="DisplayText"/>
      <xs:element name="Code" type="XAxisDomainLabelCodeType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name="XAxisDomainLabelCodeType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>

  <xs:complexType name="YAxisDomainLabelType">
    <xs:sequence>
      <xs:element ref="DisplayText"/>
      <xs:element name="Code" type="YAxisDomainLabelCodeType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name="YAxisDomainLabelCodeType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>

  <xs:complexType name="XYPlotLabelType">
    <xs:sequence>
      <xs:element ref="DisplayText"/>
      <xs:element name="Code" type="XYPlotLabelCodeType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name="XYPlotLabelCodeType">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>


</xs:schema>
```

3.6 Future Considerations

3.6.1 Viewer Design

This document focuses on the dataset. However, the data must allow support for the following viewer functions:

1) Onscreen display of the submitted waveform data.

2) Support onscreen calipers in both X and Y axis.

3) Display the annotation information both at the individual waveform level and at the group level.

4 Interfaces To Existing Data Standards

The proposed standard does not create a standalone data set. Additional context about the time-series recording will be furnished in files that are external to the XML time-series data files. The data will be linked by the creation of a unique identifier that will unambiguously identify a recording in a study. Additionally effort has been made in the design to utilize existing standards where applicable to allow for easier integration with databases and promote greater interoperability in the future.

4.1 XML Conventions

Are there XML conventions that are relevant to reference here? For example, are their conventions for specifying dates and times or other common fields or should we link them to corresponding DICOM conventions where applicable?

4.2 CDISC

The CDISC core variables, e.g. studyID, siteID, etc., are used in the *RecordingSession* entity to link to the CDISC submission data set.

Potential other ties TBD.

4.3 DICOM

Since DICOM is not based on XML, it is excluded from being used to formulate this standard. However, DICOM has many useful concepts to borrow to assist in the implementation of this standard. First, DICOM has aggregated a very complete list of lead identifiers for ECG and other domains from various standards bodies. Where this FDADF standard is required to provide a label for a plot, the design calls for the DICOM lead labeling scheme to be used whenever possible. This will allow for more meaning to be applied to the data set without reinventing another lead-naming mechanism.

DICOM has also specified a set of fiducial point markers. Again, whenever possible the DICOM list should be used to provide a consistent identification scheme.

Potential other ties TBD- for instance we should check plot attributes against the complete set of DICOM waveform attributes for completeness and applicability.

4.4 HL7 Version 3.0

HL-7 Version 3.0 is based on XML and would appear to be useful for implementing the standard. However, version 3.0 does not include a waveform description model at this time. HL-7 V3.0 does provide the generic functions of encapsulation and of reference to foreign (non-HL7) objects. Therefore, it should be possible to incorporate the proposed data set within a HL-7 message, if desired.

5 Requirements Mapping

FDA Software Development Guidelines [6] call for traceability of requirements through the design. This section TBD, but will demonstrate that the requirements presented have been satisfied.

6 Glossary

Fiducial Point: The points that begin or end a segment of the QRS. Includes P-Onset, P-Offset, QRS-Onset, QRS-Offset, and T-Offset.

Interval: A length between two fiducial points. The P-Interval is the length between P-Onset and P-Offset. The units are in samples and each sample represents *N* milliseconds.

7 References

[1] Brown B., Kohls, M., Stockbridge, N., et. al., *FDA XML Data Format Requirements Specification*

[2] Norm's 11-19-2001 proposal document.

[3] CDISC Information, www.cdisc.org

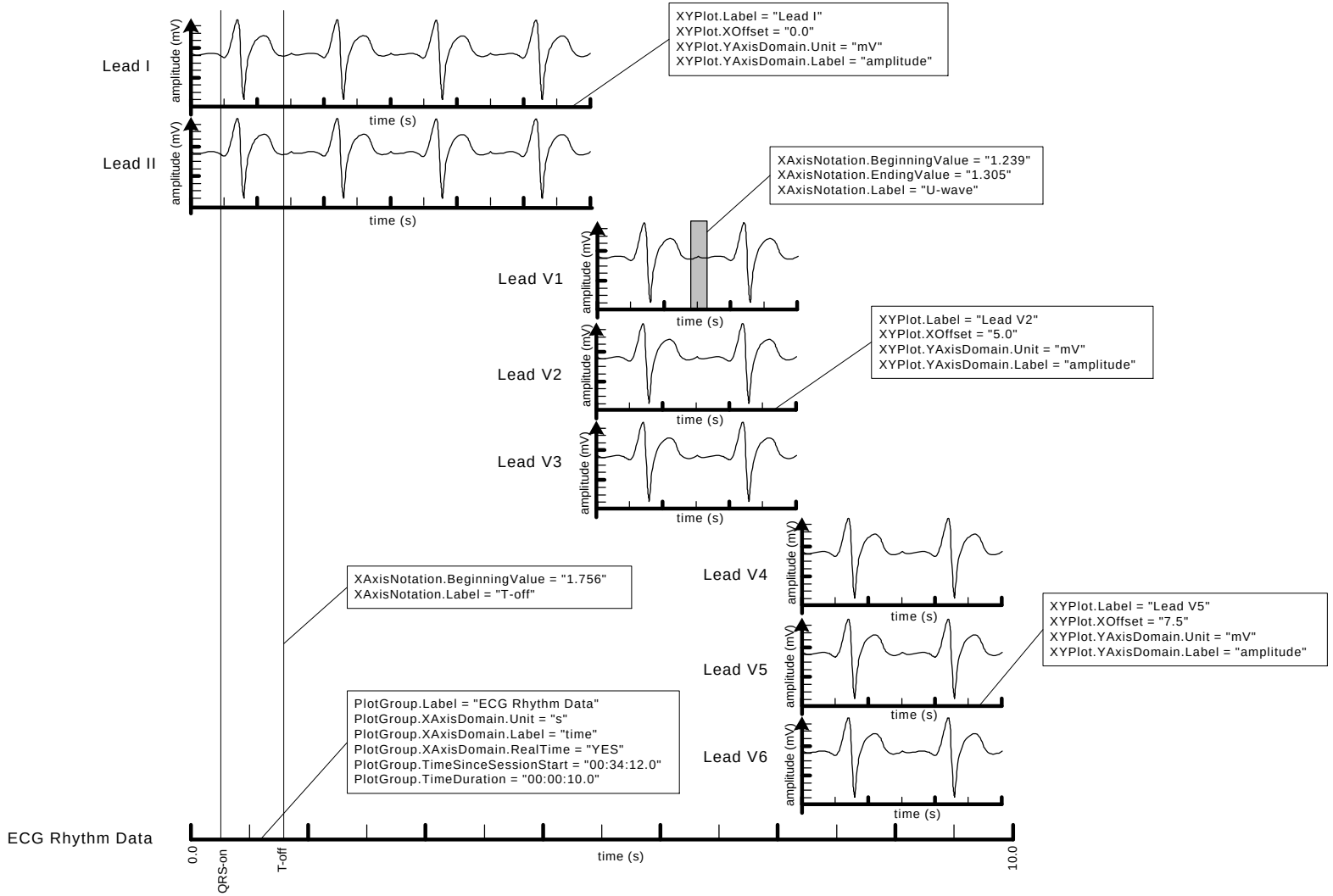
[4] HL-7 Information, www.hl7.org

[5] The FDA Website link to the 11-19-2001 presentations.

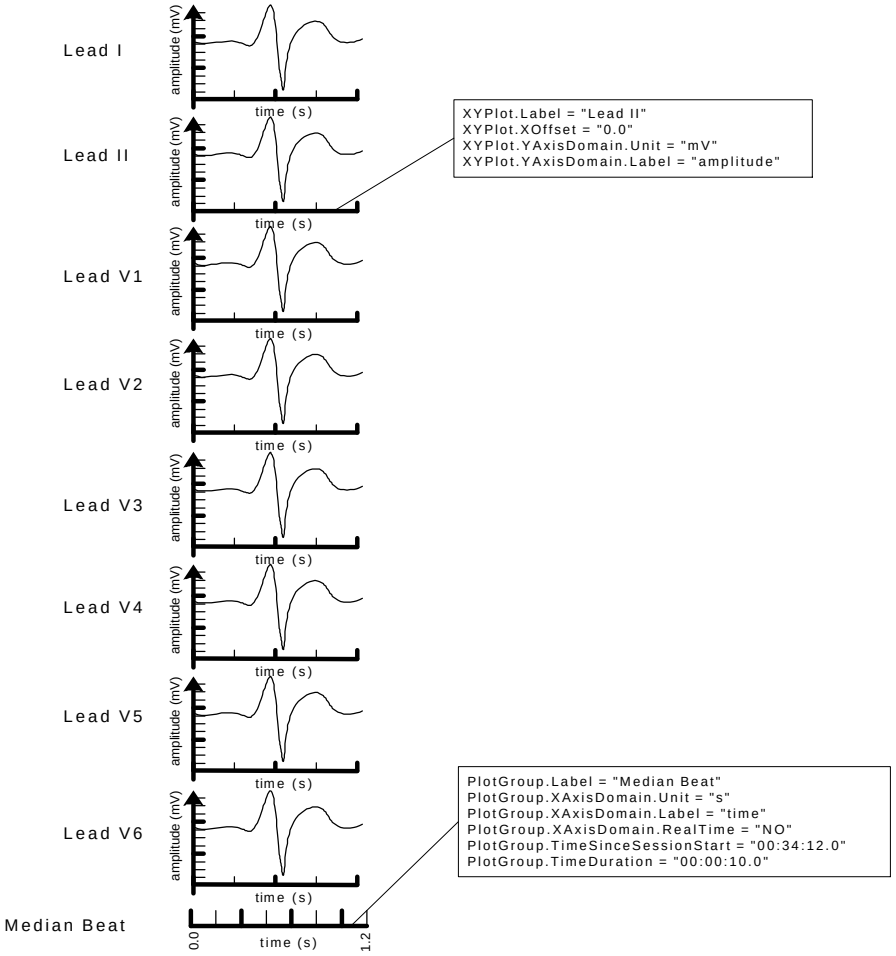
[6] FDA Software Development Guidelines

[7] DICOM Waveform Definition Supplement 30

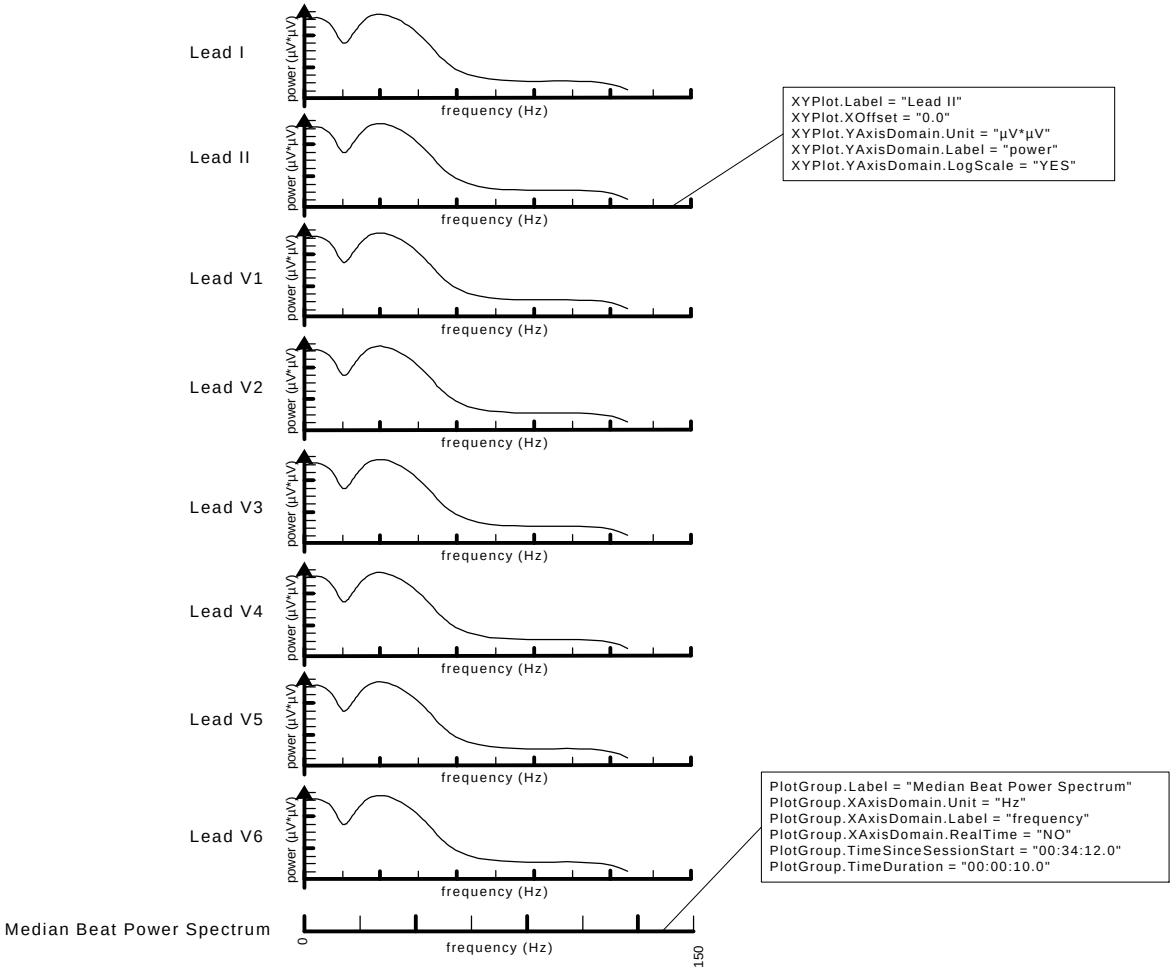
Appendix A – ECG Rhythm Data PlotGroup Example



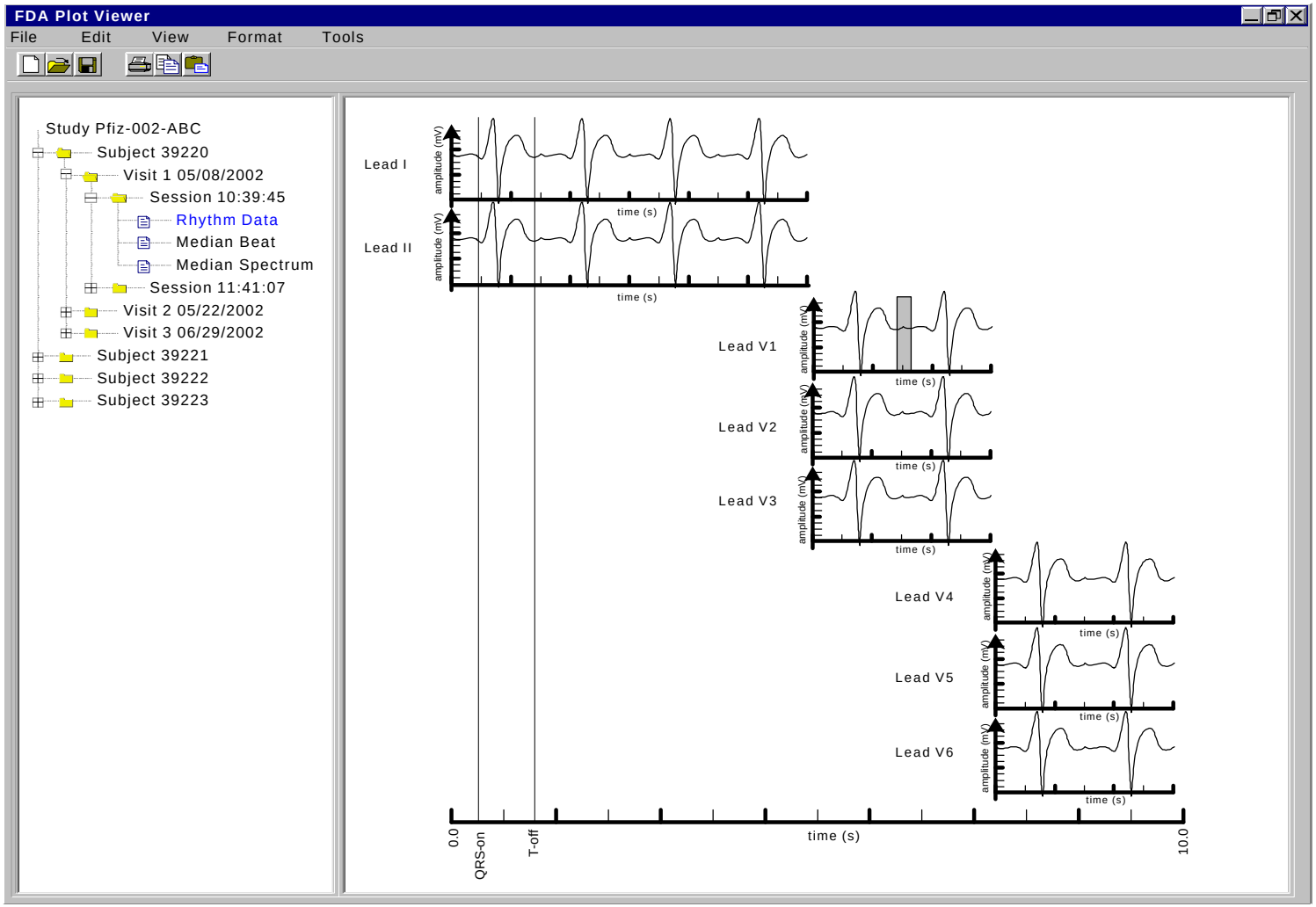
Appendix B – ECG Median Beat PlotGroup Example



Appendix C – Median Beat Spectrum PlotGroup Example



Appendix D – Viewer User Interface Concept



Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
RecordingSessionPlots	Plots from a biological signal recording made by a single device in a single session.				
StartDate	Start date of the recording. To be expressed as YYYY-MM-DD.	Yes			2002-10-24
StartTime	Start time of the recording. To be expressed as HH:MM:SS.SSS.	Yes			16:04:33.000
Duration	Duration of recording. The format is P*n*Y*n* M*n*DT*n*H *n*M*n*S, where *n*Y represents the number of years, *n*M the number of months, *n*D the number of days, 'T' is the date/time separator, *n*H the number of hours, *n*M the number of minutes and *n*S the number of seconds. The number of seconds can include decimal digits to arbitrary precision.				PT10.000S
UniqueID	Unique ID for the RecordingSession.				
FormatVersion	Version of the XML grammar.			1.0	
TrialIdentifiers	CDISC SDM metadata variables.				
StudyID	See CDISC Submission Metadata Model.				

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
SiteID	See CDISC Submission Metadata Model.				
InvestigatorID	See CDISC Submission Metadata Model.				
UniqueSubjectID	See CDISC Submission Metadata Model.				
SubjectID	See CDISC Submission Metadata Model.				
SubjectAge	See CDISC Submission Metadata Model.				
SubjectSex	See CDISC Submission Metadata Model.				
SubjectRace	See CDISC Submission Metadata Model.				
TreatmentCode	See CDISC Submission Metadata Model.				
TreatmentGroup	See CDISC Submission Metadata Model.				
Country	See CDISC Submission Metadata Model.				
VisitNumber	See CDISC Submission Metadata Model.				
VisitDay	See CDISC Submission Metadata Model.				
VisitName	See CDISC Submission Metadata Model.				
RecordingDevice	Description of the device that made the recording.				

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
Type	Type of recording device.				ECG
Manufacturer	Manufacturer of recording device.				Sam's ECG Co.
Model	Manufacturer's model name.				Super Deluxe 1000
SerialNumber	Device's serial number.				00304569381
DeviceID	ID given to the device.				34
SoftwareVersion	Device's softawre version.				10.3
PlotGroup	A group of related 2-D plots that share a common X-axis domain and can be annotated as a group (along the X-axis).				
Label	Label/Title/Name of plot group.	Yes			ECG Rhythm Data
Comment	Comments about the plot group.				Subject under stress 3 hours after dose.

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
TimeSinceSessionStart	The PlotGroup is related to a period of time relative to the RecordingSession's startTime. This indicates the offset from that point. E.g. if the PlotGroup shows 3 beats of data used for QT measurements, this would indicate the offset from the beginning of the recording to the 3 beats. The format is P*n*Y*n* M*n*DT*n*H *n*M*n*S, where *n*Y represents the number of years, *n*M the number of months, *n*D the number of days, 'T' is the date/time separator, *n*H the number of hours, *n*M the number of minutes and *n*S the number of seconds. The number of seconds can include decimal digits to arbitrary precision.			PT0S	
TmeDuration	Duration of period of time represented by this PlotGroup. E.g. if a standard 12-lead ECG, duration is 10 seconds. The format is P*n*Y*n* M*n*DT*n*H *n*M*n*S, where *n*Y represents the number of years, *n*M the number of months, *n*D the number of days, 'T' is the date/time separator, *n*H the number of hours, *n*M the number of minutes and *n*S the number of seconds. The number of seconds can include decimal digits to arbitrary precision.	Yes			PT10.0S

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
XAxisDomain	Describes the X axis domain.				
Unit	Name of base unit for X-axis. Use the Unified Code for Units of Measure (UCUM) when appropriate.				ms
Label	Name given to the X-axis.	Yes			time
MinorTickInterval	Distance between minor tick marks.				0.04 (if X-axis units of second)
MajorTickInterval	Distance between major tick marks.				0.2 (if X-axis units of second)
LogScale	Use a LOG scale?		YES, NO	NO	
RealTime	Indicates if the X-axis is in the real-time domain. If “Yes”, then the X-axis can be given values of absolute time according to the TimeSinceSessionStart attribute.		YES, NO	NO	
XAxisNotation	Annotation along the X axis.				

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
BeginningValue	Offset from the origin of the X axis to the beginning of the notation.	Yes			
EndingValue	Offset from the origin of the X axis to the end of the notation. If 0 (default), it is a point notation. If non-zero, it is an interval notation.			0	
Label	Label/Title/Name of the notation.	Yes			
Comment	Comments about the notation.				
XYPlot	A 2-D plot.				
Label	Label/Title/Name of plot.	Yes			Lead II
Comment	Comments about the plot.				
XOffset	Offset within the PlotGroup in the X domain.			0	
Connected?	Are the series of points to be connected? If not, the points will be plotted without drawing a line between them.		YES, NO	YES	
AspectRatio	Ratio of Plot.yUnit's to PlotGroup.xUnit's.			1.0	2500 (if Y-axis is units of ?V and X-axis is units of second)

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
YAxisDomain	Describes the Y axis domain.				
Unit	Name of base unit for Y-axis. Use the Unified Code for Units of Measure (UCUM) when appropriate.				uV
Label	Name given to the Y-axis.	Yes			time
MinorTickInterval	Distance between minor tick marks.				0.04 (if X-axis units of second)
MajorTickInterval	Distance between major tick marks.				0.2 (if X-axis units of second)
LogScale	Use a LOG scale?		YES, NO	NO	
XValues	The list of X values to be plotted.				
Scale	Scale factor to use to convert the X value into a PlotGroup.XAxisDomain.Unit. The scale operation happens before translation using the Offset attribute.			1.0	0.002 (if X-axis is units of second and the data is from a 500 Hz recording)

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
Offset	The number of PlotGroup.XAxisDomain.Unit's to translate the X value after scaling.			0.0	
InitialValue	If the X values are evenly spaced, they can be generated from an initial value and increment. This is the initial value. To be scaled and translated according to Scale and Offset if those attributes are present.				0
Increment	If the X values are evenly spaced, they can be generated from an initial value and increment. This is the increment to use. To be scaled and translated according to Scale and Offset if those attributes are present.				1
Values	If the X values are not evenly spaced, this is the list of X values separated by white space.				
YValues	The list of Y values to be plotted.				
Scale	Scale factor to use to convert the Y value into a XYPlot.YAxisDomain.Unit. The scale operation happens before translation using the Offset attribute.			1.0	
Offset	The number of PlotGroup.YAxisDomain.Unit's to translate the Y value after scaling.			0.0	

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
InitialValue	If the Y values are evenly spaced, they can be generated from an initial value and increment. This is the initial value. To be scaled and translated according to Scale and Offset if those attributes are present.				0
Increment	If the Y values are evenly spaced, they can be generated from an initial value and increment. This is the increment to use. To be scaled and translated according to Scale and Offset if those attributes are present.				1
Values	If the Y values are not evenly spaced, this is the list of Y values separated by white space.				
YAxisNotation	Annotation along the Y axis.				
BeginningValue	Offset from the origin of the Y axis to the beginning of the notation.	Yes			
EndingValue	Offset from the origin of the Y axis to the end of the notation. If 0 (default), it is a point notation. If non-zero, it is an interval notation.			0	
Label	Label/Title/Name of the notation.	Yes			
Comment	Comments about the notation.				

Appendix E - Data Format Field Definitions

Entity / Attribute	Description	Required	Values	Default	Example
PointNotation	A point notation with an X and Y position.				
XValue	X position of notation.	Yes			
YValue	Y position of notation.	Yes			
Label	Label/Title/Name of the notation.	Yes			
Comment	Comments about the notation.				