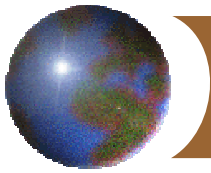


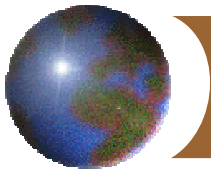
Automating the Liturgical Calendar of the Syriac Orthodox Church

Thomas Joseph, Beth Mardutho: The Syriac Institute
Simon Skaria, Microsoft



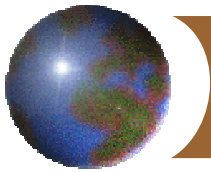
Introduction

- ◻ The pursuit of accurate liturgical calendars have inspired scientific endeavors over many centuries. This led to the globally accepted calendar in modern use, a monumental task accomplished by a commission of nine that included the astronomer and Syriac Orthodox Patriarch Neʿmet Allah I (reign 1557-1576).
- ◻ As in all mainstream Christian Churches, the liturgical calendar provides a quotidian guide for the practice of spiritual life in the Syriac Orthodox Church.
- ◻ The fullest surviving calendar appears to have been compiled by Rabban Slibo bar Khayrun (d. c. 1340), from Hah in Tur `Abdin in the early 14th century.
- ◻ Before the advent of printing, each diocese, and sometimes each church, might have its own calendar of saints whom it commemorated; hence a variety of calendars from the Middle Ages have come down to us in the manuscripts. (Brock et. al., 2001)
- ◻ Liturgical calendars have been manually compiled over the ages—a tedious task. Knowledge of the algorithms governing the generation of the calendar is limited to a few.
- ◻ The algorithms as well as design of an application that automates the calendar generation are described herein.



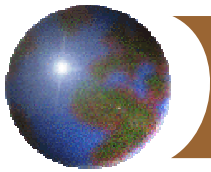
The Liturgical Calendar

- ⊕ The liturgical calendar year begins on *qudosh `idto* (Consecration of the Church) which falls on the eighth Sunday before Christmas.
- ⊕ The Syriac Orthodox liturgical year is divided into seven cycles based on the six pivot feasts below:
 - ⊕ *yaldo* (Christmas) on Dec 25th.
 - ⊕ *denho* (Epiphany) on Jan 6th.
 - ⊕ *qyomto* (Resurrection) on the first Sunday after the first full moon following the vernal equinox, variable.
 - ⊕ *m̄talé* (Transfiguration) on Aug 6th.
 - ⊕ *shunoyo* (Assumption of the Mother of God) on Aug 15th.
 - ⊕ *sleebo* (The Holy Cross) on Sep 14th.
- ⊕ Feasts and memorial days are generally celebrated on fixed dates every year. However, some feasts are based on the date of Easter which changes from year to year. Further, the Sundays in each feast cycle (that follow principal feasts) fall on different dates each year as well.
- ⊕ In the Middle East, the date of Easter is based on the Julian calendar. In Malankara (India), the date of Easter follows the Gregorian calendar.



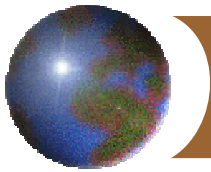
Calendar Computation

- ☉ The computation of the liturgical calendar dates in a Gregorian year requires the following steps:
 1. Determine whether the year is a leap year.
 2. Assign fixed feasts to the dates in the year.
 3. Compute date of the Gregorian or Julian Easter on the Gregorian calendar.
 4. Compute dates of movable feasts based on Easter in the following steps:
 - a. Compute dates for feasts that fall in the 10 weeks stepping back from the Easter date. The tenth Monday before Easter is the Monday of the Rogation of the Ninevites.
 - b. Compute dates for feasts stepping forward from Epiphany (Jan 6th) until the Monday of the Rogation of the Ninevites (3 day fast).
 - c. Compute dates for feasts from Easter until Transfiguration (Aug 6th).
 5. Compute dates for Sunday(s) from Transfiguration until Assumption (Aug 15th).
 6. Compute dates for Sundays from Assumption to the *sleebo* (Sep 14th).
 7. Compute dates for Sundays from *qudosh`idto* to Christmas (stepping back by week to the 8th Sunday before Christmas.)
 8. Compute dates for Sundays from *sleebo* to *qudosh`idto*.
 9. Compute date of Sunday after Christmas that falls on or before Dec 31st.
 10. Compute dates for Sundays from Jan 1st to Jan 6th referring to the day of Christmas of the previous year.



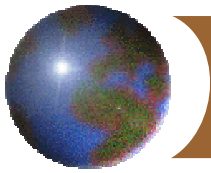
Easter Computation

- From AD 326, following the canon of the Nicene Synod, Easter was determined as the first Sunday after the Paschal Full Moon—the first full moon following the vernal equinox—of the year based on the Julian calendar.
- Since the introduction of the Gregorian calendar in October 1582 in Italy, the date of Easter has been calculated as the first Sunday after the Paschal Full Moon based on the Gregorian calendar.
- The Gregorian calendar was adopted gradually in Europe. The last country to adopt it was Greece in 1923. Orthodox Churches (including the Syriac Orthodox) now calculate the date of Easter based on the Julian calendar and convert to the equivalent Gregorian date.
- In 1952, Patriarch Aphrem I authorized the observance of Easter according to the Gregorian calendar in Malankara (India) and its diaspora (from 1953).
- The Easter calculation implemented is based on the Ron Mallen algorithm published at <http://users.chariot.net.au/~gmarts/eastalg.htm>.
- The algorithm applies for the years 1583 (when the Gregorian calendar was introduced) to 4099 (when the Gregorian calendar will require a day's adjustment to remain accurate).



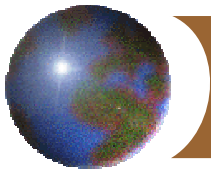
Assigning Lections for Feast Days

- ⊕ The lectionary prescribes the scripture readings for Sundays and feast days. Readings are listed for the canonical hours of evening and morning, before and during divine liturgy, and for the special offices of certain feast days.
- ⊕ A date may have none, one, or more than one feast.
- ⊕ For dates with more than one feast, order feasts in the descending order of priority.
- ⊕ Retrieve and assign lections of the offices for each date with at least one feast date based on the feast with the highest priority.
- ⊕ Lections from a lower priority feast may be assigned for an office if the higher priority feast on that date does not have a corresponding lection for that office. These rarely occur.



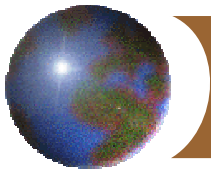
Implementation Overview

- ➊ Algorithms are implemented in a Java application.
- ➋ Feasts and lectionary data are persisted in an Access database; accessed from the Java application through JDBC.
- ➌ Java application outputs calendar and lection data into an XML file.
- ➍ XSLT transforms the XML into HTML for display. The same XML file can be transformed to produce a calendar, a lectionary, or any variation thereof, with appropriate XSLT transformations.



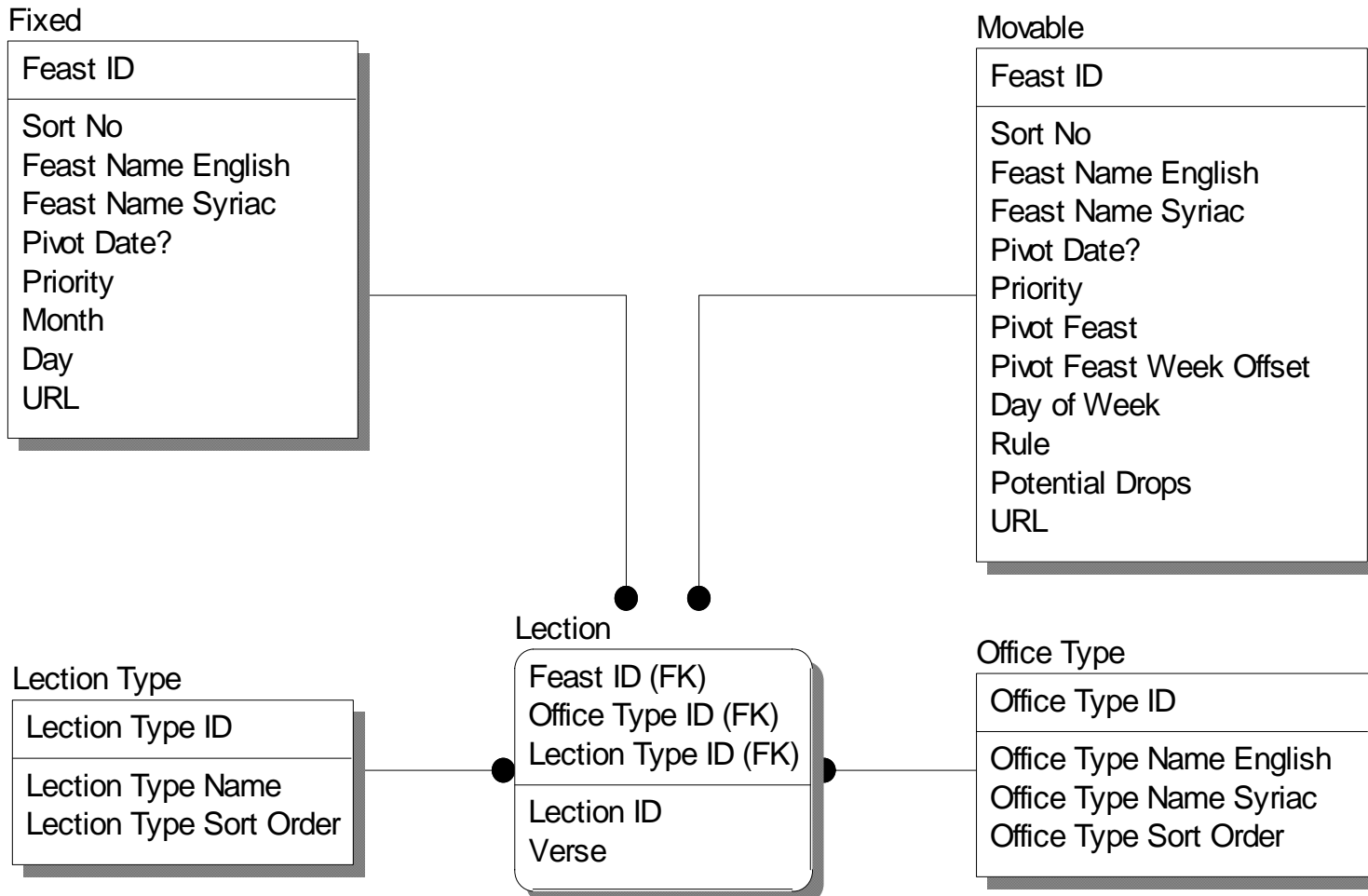
Implementation - The Inputs

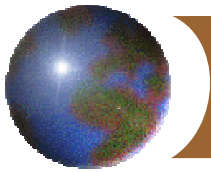
- ➊ The Gregorian Year for which the liturgical calendar is to be generated.
- ➋ Calendar type for Easter computation – Gregorian or Julian.
- ➌ The feasts and lectionary data source (Access database file).
- ➍ Output XML file name.



Implementation - The Feasts & Lectionary Database

📍 Data Model



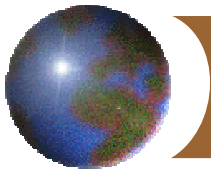


Implementation - The Feasts & Lectionary Database

Fixed Feast – Sample Records

Sort No	1061	1151
Feast ID	FDEJA06	FVMJA15
Feast Name English	Epiphany	Intercession to BVM for Seeds
Feast Name Syriac	<i>denho</i>	-
Pivot Date?	Y	N
Priority	1	3
Month	Jan	Jan
Day	6	15
URL	http://sor.cua.edu/Feasts/Epiphany.html	http://sor.cua.edu/Feasts/VirginMary.html

- Five fixed feast records—Epiphany, Transfiguration, Assumption of the Mother of God, Festival of the Holy Cross, and Christmas—have Pivot Date? set to “Y”.
- The highest priority of 1 is assigned to all *moronoye* (feasts of the Lord—Circumcision, Epiphany, Presentation at the temple, Transfiguration, and Christmas), St. John the Baptist, St. Stephen, Annunciation to the Virgin Mary, and the festival of the Holy Cross. Locally commemorated memorials have the lowest priority.
- Memorial days can be added to the fixed feast table.



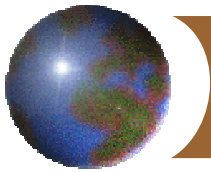
Implementation - The Feasts & Lectionary Database

📍 Movable Feast – Sample Records

Sort No	1	15
Feast ID	MSUC-08	MSUD+05
Feast Name English	Consecration	5 th Sunday after Epiphany
Feast Name Syriac	<i>qudosh `idto</i>	-
Pivot Date?	N	N
Priority	1	5
Pivot Feast	FCHDE25	FDEJA06
Pivot Feast Offset	-8	5
Day of Week	Sun	Sun
Rule	8 th Sunday before Christmas	5 th Sunday after Epiphany
Potential Drop	N	Y

URL

- ❑ The only movable feast which is a pivot date is the *qyomto* (Easter).
- ❑ Lectionary has readings for Sundays after major feast days which are hence listed as movable feasts. Some of these Sundays may not occur in a given year if pre-empted by the next feast cycle.

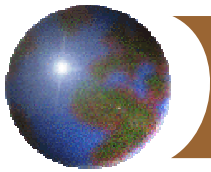


Implementation - The Feasts & Lectionary Database

Lection – Sample Records

Lection ID	Feast ID	Office	Type ID	Lection Type ID	Verse
10101	MSUC-08	EVE		GSP	St. Matthew 16:13-23
10102	MSUC-08	MOR		GSP	St. John 21:15-22
10103	MSUC-08	BQU		OT1	Exodus 33:5-11
10104	MSUC-08	BQU		OT2	I Kings 6:1-8
10105	MSUC-08	BQU		OT3	Jeremiah 31:10-14, 22-26
10106	MSUC-08	BQU		OT4	Ezekiel 9:11, 10:5, 18-22
10107	MSUC-08	QUR		GE	Acts 7:44-53
10108	MSUC-08	QUR		SP	Hebrews 9:1-14
10109	MSUC-08	QUR		GSP	St. Mark 8:27-33

- Lectionary entries are recorded for each office of each feast.
- Most of the feasts have readings for the Gospel of *ramsho* (vespers), Gospel of *saphro* (matins), Old Testament readings before the divine liturgy, and the General Epistle, Epistle of St. Paul, and the Gospel readings during the public celebration of the divine liturgy.
- Some feasts have special offices. e.g. the Gospel readings at the fire pit during Christmas services, blessing of the palm leaves on Palm Sunday, during the three services of the Pentecost, etc.



Implementation - Classes

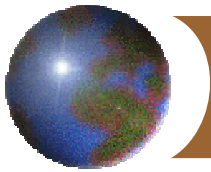
✚ calGen

☒ Attributes

- Gregorian Year
- Calendar Start Date
- Year Vector (length=# of days in the yr)
- Index of Easter Date
- Julian Easter Calculation Indicator [Optional]
- Easter Calculation Override Indicator & Easter Month/Date [Optional]
- ODBC Database String
- Database Connection Object

☒ Methods

- main(String args[])
- setAllFixedFeasts()
- setEaster()
- stackOfPreEaster()
- stackOfPostDenho(mondayMinus10)
- stackOfPostEaster()
- stackOfPostTransfig()
- stackOfPostAssumption()
- stackOfPreXmas()
- stackOfPostSleebo(sundayMinus8)
- stackOfPostXmas()
- stackOfPostXmasPrevYear()
- prioritize(feast startFeast)
- readLection()
- genXML(strOutputXMLFile)



Implementation - Classes

📅 calendarDay

📦 Attributes

- Gregorian Calendar Date
- feast
- lection
- Number of feasts

📅 feast

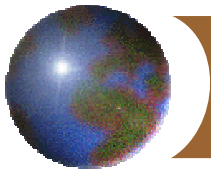
📦 Attributes

- is a Fixed feast
- is a Pivot feast
- Feast ID
- Pivot feast (on which the Feast is based)
- English feast name
- Syriac feast name
- URL for the feast
- Feast priority
- **Next feast (on the same date)**

📅 lection

📦 Attributes

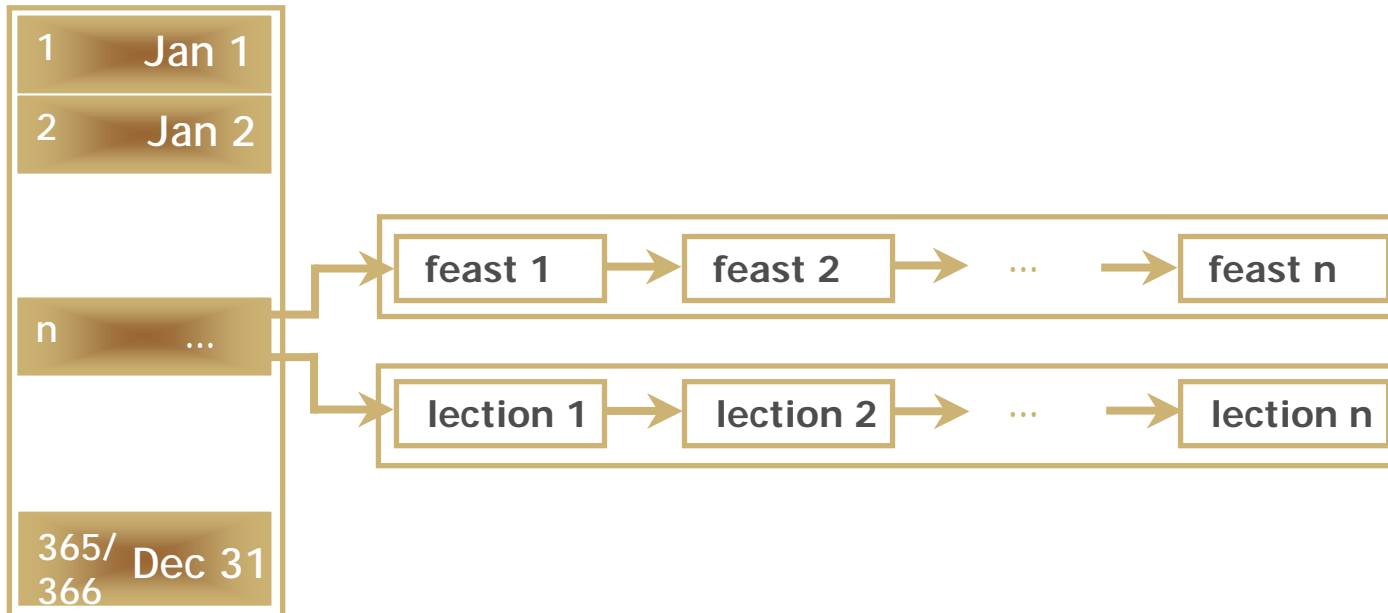
- Office type
- Office type name
- Lection type
- Lection type name
- Verses
- ID of feast contributing the lection
- **Next lection**

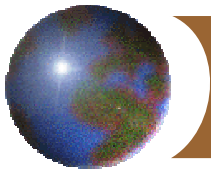


Implementation - Populating the Data Structures

- Initialize the yearVector array of length equal to number of days in the year.
- Assign dates (month & day of the month) to the array cells from Jan 1 to Dec 31.
- Instantiate feast objects for each applicable date. Order feast objects by priority.
- Instantiate lection objects for each office based on the highest priority feast. Add lections for any office that applies to a lower priority office alone.

yearVector





Implementation - Data Structures Example - June 29, 2003

argYear = 2003
yearVector[365]

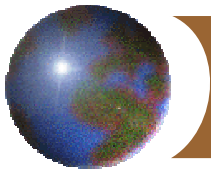
i = 179
calendarDay
date = Jun 29
feast
lection
numFeasts = 2

feast 1
is a Fixed feast = "Y"
is a Pivot feast = "N"
Feast ID = "FAPJN29"
Pivot feast = null
English name = "Apostles Peter & Paul"
...
Feast priority = 2
Next feast = feast 2

feast 2
is a Fixed feast = "N"
is a Pivot feast = "N"
Feast ID = "MSUQ+10"
Pivot feast = "MSUQ+00" [Easter]
English name = "3 rd Sunday after Pentecost"
...
Feast priority = 5
Next feast = null

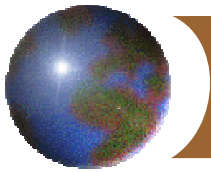
lection 1
Office type = "Evening"
Lecture type = "Gospel"
Verses = "St.Matthew 16:13-20"
Contributing Feast ID = "FAPJN29"
Next lection = lection 2

lection 10
Office type = "Divine Liturgy"
Lecture type = "Gospel"
Verses = "St. Matthew 20:1-16"
Contributing Feast ID = "FAPJN29"
Next lection = null



Implementation - The XML Output

```
C:\Documents and Settings\tjosephlocal\My Documents\web\calGen\cal.xml - Microsoft Internet Explorer provided by Cap Gemini Ern
File Edit View Favorites Tools Help Google Links
<?xml version="1.0" ?>
<!DOCTYPE calendar (View Source for full doctype...)>
- <year val="2004">
- <month id="1" nm="January">
- <week id="1">
  <day id="1" nm="Sun" />
  <day id="2" nm="Mon" />
  <day id="3" nm="Tue" />
  <day id="4" nm="Wed" />
- <day id="5" nm="Thu" numfeasts="2">
  <date>1</date>
  <feast id="FCIJA01" priority="1">Circumcision of our Lord; New Year</feast>
  <feast id="FBGJA01" priority="4">Baselius of Caesarea (c. 329-379), Gregory of Nyssa (c. 330-395), all holy
    fathers and teachers</feast>
  + <office nm="Evening">
  + <office nm="Morning">
  + <office nm="Before Qurbono">
- <office nm="Divine Liturgy">
  <lection nm="General Epistle">Acts 7:2-10</lection>
  <lection nm="St. Paul">Romans 2:28-3:8</lection>
  <lection nm="Gospel">St. John 15:8-19</lection>
  </office>
</day>
- <day id="6" nm="Fri" numfeasts="0">
  <date>2</date>
</day>
- <day id="7" nm="Sat" numfeasts="0">
  <date>3</date>
</day>
</week>
+ <week id="2">
+ <week id="3">
+ <week id="4">
+ <week id="5">
</month>
+ <month id="2" nm="February">
+ <month id="3" nm="March">
```



Implementation - XML Document Type Definition (DTD)

<!ELEMENT year (month)+ >

<!ATTLIST year

 val ID #REQUIRED >

<!ELEMENT month (week)+ >

<!ATTLIST month

 id ID #REQUIRED

 nm (January | February | March | April | June | July | August | September | October | November | December)
 #REQUIRED >

<!ELEMENT week (day)+ >

<!ATTLIST week

 id ID #REQUIRED >

<!ELEMENT day (date | feast* | office*) >

<!ATTLIST day

 id ID #REQUIRED

 nm (Sun | Mon | Tue | Wed | Thu | Fri | Sat) #REQUIRED >

<!ELEMENT date (#PCDATA) >

<!ELEMENT feast (#PCDATA) >

<!ATTLIST feast

 id ID #REQUIRED

 priority CDATA #IMPLIED >

<!ELEMENT office (lection+) >

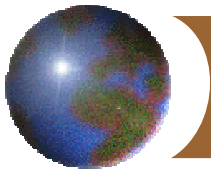
<!ATTLIST office

 nm CDATA #REQUIRED >

<!ELEMENT lection (#PCDATA) >

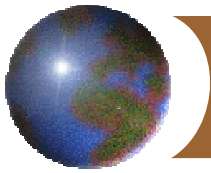
<!ATTLIST lection

 nm CDATA #REQUIRED >



Implementation - XSL Transformations of the XML data

- ➊ XSLT implemented for transforming XML into an HTML document presenting the data in a traditional calendar format.
- ➋ A calendar page is displayed in two sections.
 - ▣ Dates in the month are displayed in a tabular format. Significant feasts are listed under the corresponding dates. The Gospel readings of the Divine Liturgy are listed for Sundays and feast days.
 - ▣ A footer section lists all feasts including minor feasts and memorial days.
- ➌ Each date node of the XML document is traversed through twice to generate the two sections. The attributes of the feast and lection nodes are tested in the first pass to select only high priority feasts and Gospel lections of the divine liturgy for display in the calendar section.
- ➍ Alternate XSLTs can transform the data into a lectionary style presentation as well as other formats such as rtf for printing.



Future Enhancements

- ➊ Convert Java application to a servlet with a JSP front end that submits input parameters and generates the calendar online.
- ➋ Port application to .NET environment.
- ➌ XSLT transformation to Rich Text Format for print editions of the calendar.
- ➍ XSLT for generating a lectionary for a calendar year.