

# 卒業論文

## 古今の星座の違いを示す星図アプリケーション

大阪工業大学  
情報科学部 情報メディア学科  
C03-044 河津秀明

2008年2月29日

# 目次

1	序論	4
1.1	背景	4
1.2	古墳の概要	4
1.3	試み	4
2	現在の星座の起源	6
2.1	現在の星座	6
2.2	起源	6
2.3	メソポタミアからギリシャへ	9
3	古代日本の星座の起源	13
3.1	古代中国が起源の星座 星宿	13
3.2	古代中国の地上と天上	14
3.3	日本へと伝わる星座(星宿)	15
4	星宿図解説 - 高松塚古墳の天井画から	17
4.1	天井画・壁画の意味	17
4.2	分野説について	18
4.3	古代中国の星宿と西洋現代の星座の対応	19
5	星図アプリケーション解説	21
5.1	アプリケーションの概要	21
5.2	データベース構築	21
5.3	使用した古代・現代の星座	23
5.4	メルカトル座標	28
5.5	円座標	31
5.6	各操作ボタン	32
5.6.1	星描画ボタンと機能の解説・工夫	32
5.6.2	星座線ボタン	33
5.6.3	名前表示ボタン	33
5.6.4	特定の星座拡大ボタン	34
5.7	星座名の抽出・配置・表示	35
5.8	拡大表示機能	36
5.9	星座線描画機能の解説	37
5.9.1	星の同定方法	37
5.9.2	座標ごとの対応	38
5.10	Flash8 でのデータの取り扱い	38
6	比較による結果	40
7	まとめ	46

8	參考資料	47
9	付録	48

# 1 序論

## 1.1 背景

本研究では、日本にて高松塚古墳やキトラ古墳などで発見された星座と現代の星座との対応比較を容易に行えるアプリケーションの開発を行った。高松塚古墳とキトラ古墳には同じ形の星座が多いため、比較のために古代中国の星座も表示できる機能も備えている。古代中国の星座はアスターナ古墳群の墳墓の一つから発掘された星座を参考にしている。

このアプリケーションを開発するにあたり、以下の3点を中心のテーマとした。

- 古天文学を通して自国の文化への関心を高めて欲しい。
- これからの未来を担う小・中学生のために何かを残したい。
- 教育教材のアプリケーションで児童教育に貢献したい。

その他にも、使いやすさや見易さにも配慮し開発を進めてきた。

## 1.2 古墳の概要

今回、アプリケーション開発時に、日本で最も代表的な高松塚古墳とキトラ古墳の星宿図を参考にした。それぞれについて概要を以下に説明する。

高松塚古墳は、奈良県明日香村（国営飛鳥歴史公園内）に存在し、694～710年にかけて築造された古墳で、直径23m（下段）及び18m（上段）、高さ5mの二段式の円墳である。

キトラ古墳は、奈良県高市郡明日香村にある円墳で、704年～8世紀初めに建造された。上段が直径9.4m、高さ2.4m、テラス状の下段が直径13.8m、高さ90cmの二段築成作りの円墳である。墳丘は小高い阿部山の南斜面に位置している。円墳という古墳の形体や四方に四神が描かれた壁画が残されているという類似点から、高松塚古墳とは兄弟と呼称されることもある。

古代中国の星座を参考に使った古墳は、トルファンのアスターナ古墳群の中の一つである三八号墓の壁画である。アスターナ古墳群は高昌故城に住んでいた魏氏高昌・唐代（618～907年）の貴族の墓地である。高昌故城の北方約1km、火焰山の南麓にあり、500基近くの墳墓が並んでいる。

## 1.3 試み

本研究では、古代日本の星座と現代の星座とを簡単に比較することが出来る星座盤アプリケーションの作成を行った。天文学に興味がある人も、そうでない人にも古天文が容易に理解できるものを作成するように心がけた。今回アプリケーション開発に使用したツールは、Adobe社の「Macromedia Flash8」というソフトウェアを使用した。このソフトウェアではFlashムービーを作成することが出来る。このムービーの利点は、

- 低容量かつ視覚効果のあるファイル作成が容易
- Webで公開が容易

- 一般的に普及している PC でほとんどのユーザーが使用可能である
- 利用時の操作が容易

の4点を挙げる事ができる。簡単に利用可能なアプリケーションを作成することを目的としているため今回このツールを使うこととした。

今回、天文学を通して日本の文化に触れるということを目的としているので、誰にでも扱うことの出来るアプリケーションの作成を目指す。特に、小学生の高学年から中学生に古代日本について触れてもらいたいと思い、メインターゲットにしている。基礎知識がないユーザーをターゲットして想定しているので、見やすく、操作しやすいものの作成を心がけた。

## 2 現在の星座の起源

本章では、現代日本にて日常的に使われている星座がどこからどのようにして伝わったのかを歴史的観点から説明する。本章では、文献 [2] を参考文献としている。

### 2.1 現在の星座

現在、日本で使われている星座はギリシャに起源をもつものである。世界的には、さまざまな理由で星座が乱立したものを統合、整理し、1928年の国際天文学連合（IAU）第3回総会で現在の88星座が決められた。後に日本で、88の星座とそのラテン語での正式名が決まった。1944年に学術研究会議（現日本学術会議）が訳名を決定するとこれが全国的に使われるようになり、その後、数度改定され、現在に至っている。

### 2.2 起源

現在の日本で使われている星座の原型は、メソポタミアで考えられたとされている。星座の存在を直接示す最古の証拠は、メソポタミアで発掘された石碑と粘土板に見つかった。バビロンから出土した『夜の神々への祈り』と呼ばれる文書は紀元前1700年ごろに記されたもので、4つの星座と3つの星、プレアデス星団が記されている。紀元前1300年以前には、メソポタミアの人々が認識していた星はごくわずかだった。紀元前1300年以降は、境界石や円筒印章に太陽や月や惑星を表す符号と組み合わせて、これらの象徴が描かれるようになり、これらが星座を表していたのであろう。象徴が描かれた境界石を図1、図2に示す。



図 1: 星座表現された境界石 1、紀元前 1300 年頃、メソポタミア (出典: [2])

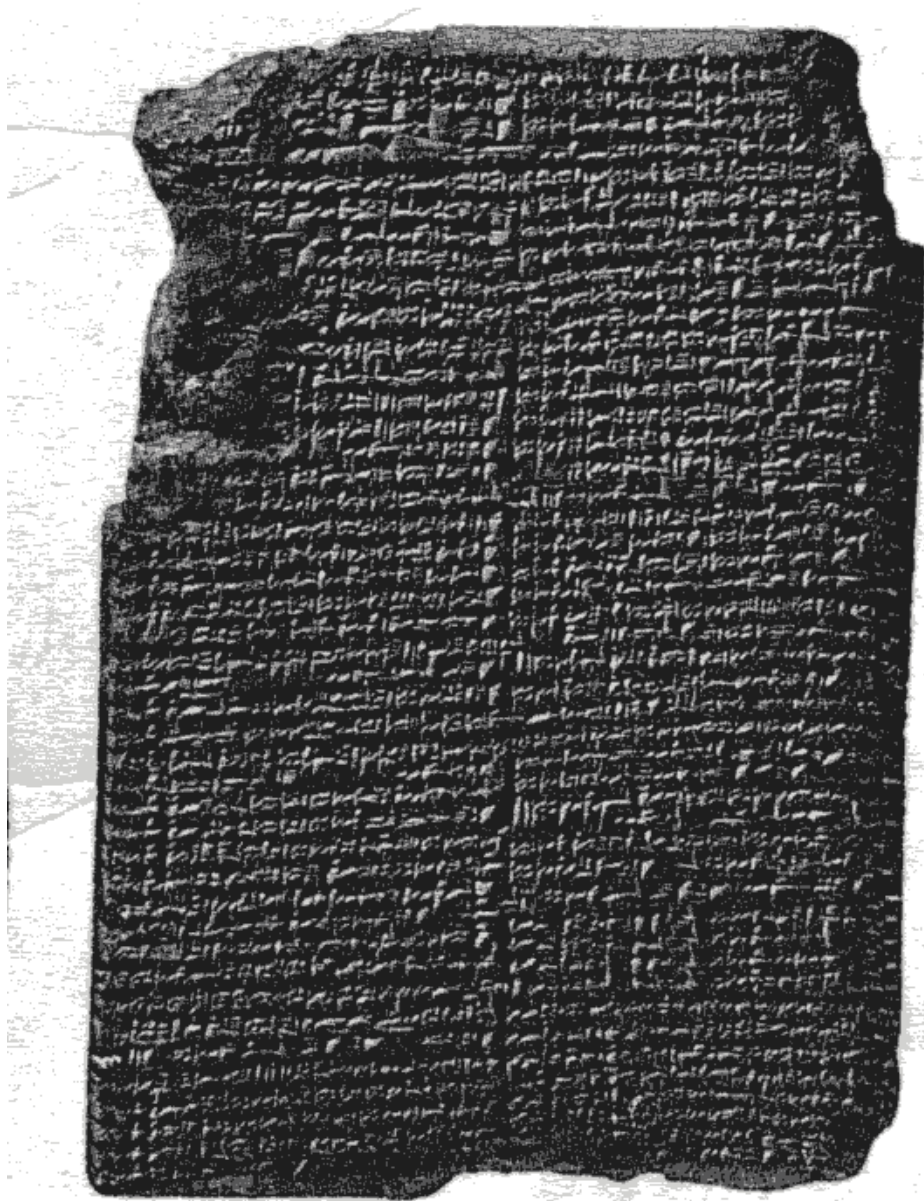


図 2: 観測結果・星座が記された境界石 2、紀元前 1100 年頃、メソポタミア（出典：[2]）

紀元前 1300 年頃から、粘土板に空の赤道帯を中心とする 3 帯域にある 30 以上の星座の名が楔形文字で刻まれ、『ムルアピン』と呼ばれる粘土板にはメソポタミアのほとんどの星の位置と動きを観察した結果が記されている。この他にも、星座の伝承の成立年代は天の歳差運動に基づく方法で求めることができる。ムルアピンでは星の天球上の相対的な位置が示されているため、情報をおよその年代に置きかえられる。例えば、春分点が今日のおひつじ座にあたる星座のすぐ東にあると記されているところから、紀元前 1000 年代の終わりごろの記述と分かる。

観測結果の多くは観測者の位置の緯度によるため、徹底分析すれば、観測地の緯度も特定できる。ウィーン大学のフンガーとブラウン大学の故ピングリーは、プトレマイオスが記述した星の出現に関する記録と比較し、ムルアピンのいくつかの星座を分析した。結果、ムルアピンの星座は紀元前 1000 年頃の北緯  $36^{\circ}$  の位置で、観測者はメソポタミア北部のアッシリアで観



測していたことが判明した。ムルアピンはメソポタミアのほぼすべての星を網羅しているので、紀元前 1000 年より前に成立していたと考えられる。メソポタミアの星座の大部分は、紀元前 1300 年から紀元前 1000 年の比較的短い期間で作られたものと分かる。また、別の分析手法によって、北半球から見える南限の星座が作られた年代と場所を知ることができる。北半球の中緯度からは、天の南極を中心とする一定の範囲内にある星は見えないということに基づき計算する方法である。見えない部分の円の中心である南極の位置を特定できれば成立年代が分かり、円の半径から星座を作った人々がいた地域が分かる。この考え方をういて、その当時に南限に見えていた 6 つの星座が紀元前 1000 年頃に北緯 33 °付近で作られたとわかった。これらのデータが矛盾しないので、メソポタミアの星座と観測データは、紀元前 1300 年頃から紀元前 1000 年頃に北緯 33 ~ 36 °の付近でアッシリア人によってもたらされたと考えてよいだろう。

ムルアピンに記されている星座の題材には、様々なものが混ざり合っている。神、動物、日常使っている農具などだ。星群をベースとした数多くの予言と、それを利用した暦が記されている。暦は農耕社会には不可欠で、予言も神も暦も神官の仕事となる。よって、星座が神官によって作られたということを示す有力な手がかりとなっている

## 2.3 メソポタミアからギリシャへ

古代ギリシャの星座にはアッシリアと同じ名称で呼ばれているものがいくつも存在する。ギリシャのやぎ座とふたご座の星々は、アッシリアでは『半山羊半魚』、『大きな双子』という類似名称で呼ばれていた。合計 20 の星座がアッシリアの名称をそのまま模している。10 の星座は構成する星が同じで名前が異なっている。アッシリアで『雇夫』、『ツバメ』が牡牛、魚と名づけられた。表 1、表 2 に示す。

表 1: 類似名称の星座 (一部) (<http://www.kotenmon.com/str/mulapin.htm> より)

ムルアピンでの星座名	西洋星座対応位置
大きな双子	双子座 , と南北の星
小さな双子	双子座 , と南北の星
クラブ	かに座
ライオン	しし座
天の牛	おうし座
Anu の真の羊飼い (王)	オリオン座
蛇	うみへび座
烏	からす座, コップ座
鷲 (わし)	鷲座のほとんど
魚	みなみのうお座
さそり	さそり座
半山羊半魚	やぎ座

表 2: 構成する星が同じで名称が異なる星座 (<http://www.kotenmon.com/str/mulapin.htm> より)

ムルアピンでの星座名	西洋星座対応位置
老人	ペルセウス座
杖 (羊飼いのつえ)	ぎょしゃ座
SU.PA	うしかい座
高官	かみのけ座
ワゴン	おおぐま座
天のワゴン	こぐま座
雌やぎ	こと座
雇夫	おひつじ座
おんどり	うさぎ座
偉大な星	みずがめ座

ギリシャ人がメソポタミアの星の情報をどのように取り入れたかは、さんかく座の例を見ればよく理解することができる。ムルアピンでは、さんかく座はおおぐま座にも用いられた『鋤』という名称だった。幾何学は、メソポタミアとエジプトの学者にはよく知られていたが、一般的に土木工事の設計技術として認識されていた。ギリシャには哲学者タレスによって取り入れられ、幾何学は実用的な小さな経験則から美と普遍性を追求する理論体系へと姿を変え、紀元前300年頃、ユークリッドが『原論』を著した。この変化があり、三角形が幾何学の礎として天に置かれたのであろう。ここから、紀元前6世紀から紀元前4世紀に、メソポタミアの星座がギリシャ人によってさんかく座と改名されたと分かる。これと同時に他の星座もギリシャに取り入れられたかどうかは不明であるが、現在明らかになっている情報から判断する限り、その可能性は低いであろう。

ギリシャの最古の文献、ホメロスの叙事詩とヘシオドスの農事暦は、プレアデス星団とアデス星団、オリオン座とおおぐま座、シリウスとアルクトゥルスについては記述しているが、それ以外の星には触れられていない。紀元前500年以前のギリシャの他の文献には、星について書かれたものは存在しない。これより、ギリシャ人は紀元前500年くらいまではごく限られた星座だけ認識していたということになる。紀元前4世紀のエウドクソスの著作で初めて星座について解説された。エウドクソスの記述には多くの情報が含まれている。例えば、『(りゅう座の)頭は星が沈む場所と昇る場所が交わるあたりをめぐっている』と記されているのは、りゅう座の頭部の星々が地平線に沈まずに、北の地平線ぎりぎりをかすめる位置に見えるという意味だ。これ単独では有効な証拠とはなりえないが、エウドクソスの記述から緯度と年代の影響を受ける部分を拾い出し分析した結果、最終的な誤差は緯度で $0.9^\circ$  (100km)以内、年代で80年以内となった。エウドクソスの記述は紀元前1130年に北緯 $36^\circ$ 、アッシリアで行われた観測であると分かった。エウドクソスの著書から得られた観測の時と場所はムルアピンのものと一致するので、どちらも紀元前1100年頃にアッシリアで行われた観測を元にしてしていると分かった。双方とも古代星座の大部分を網羅しているところから、この頃までに星座体系はほぼ完成していた。ギリシャにメソポタミアの星座が伝わった時期は、星座の完成後からエウドクソスの著書が記される紀元前366年までの間だということが分かる。

バビロニアの黄道十二星座は紀元前400年頃にギリシャに伝えられたことも文献から明らかになっている。しかし、メソポタミアからギリシャへは、どのようにして星座が伝わったのかは解明されていない。ギリシャの星座体系はメソポタミアの星座によるものが多いが、他の地域からでも起源が見出せない星座が18個存在する。これらの星座はギリシャ人が独自に生み出した新しい星座である。ギリシャにとっての星座の意味は、伝説の英雄や動物の物語を表し、暦や公開のための道具となっていくた。黄道十二星座も天球を12等分し、惑星の位置を示す黄道十二宮という座標になった。

星座の利用変遷は、天文学の一面が近代科学的に変化する様子を示している。星座は宗教的なものから民族的なものへ、実用的なものへ、最終的には科学的なものへと変化してきた。その過程で精神性が失われ、定量化が進んだ。以上の流れを時代背景も踏まえ表3に示す。

表 3: ギリシャの星座成立年表

年代	出来事
紀元前 1100 年頃	アッシリアで古代星座の大部分を網羅した観測が行われる
紀元前 8 世紀	叙事詩・農事暦にオリオン座・おおぐま座・プレアデス星団・ ヒアデス星団・シリウス・アルクトゥルスのみ記される
紀元前 585 年	タレスにより幾何学が取り入れられる
紀元前 4 世紀	エウドクソスの著作で初めて星座について解説された
紀元前 400 年頃	バビロニアの黄道十二星座が伝わる
紀元前 300 年頃	ユークリッドの原論が著される
	幾何学の礎として三角形が天に置かれる
紀元前 6 ~ 4 世紀	三角形がさんかく座に改名

### 3 古代日本の星座の起源

古代日本で使われていた星座は、古代中国が起源の星座が渡来したものであった。これは星宿と呼ばれ、古代日本で使われていた。この星宿の起源、由来について説明する。本章では、文献 [4] を参考にしている。

#### 3.1 古代中国が起源の星座 星宿

中国では農業の発達と共に、目立った星や星座は季節の目印として作られ、天体の運行を把握する基準ともされた。『書経』(960-1127年)の『堯典』に、春分・夏至・秋分・冬至の日暮れに南中する星として、それぞれを鳥・火・虚・昴という名の星を挙げている。これらは堯典の四中星と呼ばれ、現代の星座では順にうみへび座、さそり座、みずがめ座、おうし座のプレアデスに対応し、二十八宿につながるものである。二十八宿とは28の星座からなる、天の赤道帯を不等間隔に28の宿に分割した二十八宿という体系付けされた星宿の一部である。堯典は、紀元前2000年頃の天文現象を反映している。多くの星が観測されるようになり、目立たない星からも星座が作られるようになった。五経の一つ『礼記』の月令篇に二十八宿のうち毎月太陽が宿る星座、日暮れと夜明けに南中する星座がそれぞれ記されている。類似の記事は他の文献にも見られ、1938年能田忠亮はそれらを分解し、二十八宿体系の成立を紀元前620-100年頃と推定した。その後1977年に湖北省随県の曾侯乙の墓から、二十八宿名が全て記された漆器が出土した。副葬品に紀元前433年にあたる年号が記されていたので、成立が前5世紀末以前であることが確かめられた。

二十八宿の数字の由来は2説存在する。一つは月が天を一周する周期が27.3日だからだという説と、土星が見かけ上、天を一周する周期が当時は28年と考えられていたので、土星が年毎に宿る星座だと考える説である。いずれの説も、二十八宿の各星座の大きさが極端に不揃いなこと、黄道・赤道にずれがあるたことで特定することができない。また、各宿の星座の多くが目立たない暗い星からなっていることも不可解である。二十八宿は7宿ずつ東方宿・北方宿・西方宿・南方宿に分け、青竜・玄武・白虎・朱雀にそれぞれ見立てられた。各宿で代表星(距星)を定め、それらを通る赤経線で天を不等間隔に分割し、距星を基準とし、太陽や月、惑星の運行や、他の恒星の天球座標が定量的に扱われるようになった。各宿にはそれぞれ地上の地方と対応させ、地上で何かが起こると天域に異変があるとされ、逆に特異な天文現象が起こると地上に変異が起こると考えられていた。中でも、彗星や新星、超新星や日食、太陽黒点は特に重要視され記録が多く残っている。また司馬遷が紀元前100年頃に著した『史記』の『天官書』に星座と星占いについて述べた章がある。これに天の星も地上と同様に天帝を頂点とする中央集権の官僚組織体制の社会になっており、星座は天官や星官と呼ばれた。上述に対応する宋代の天文図は図3に示す。

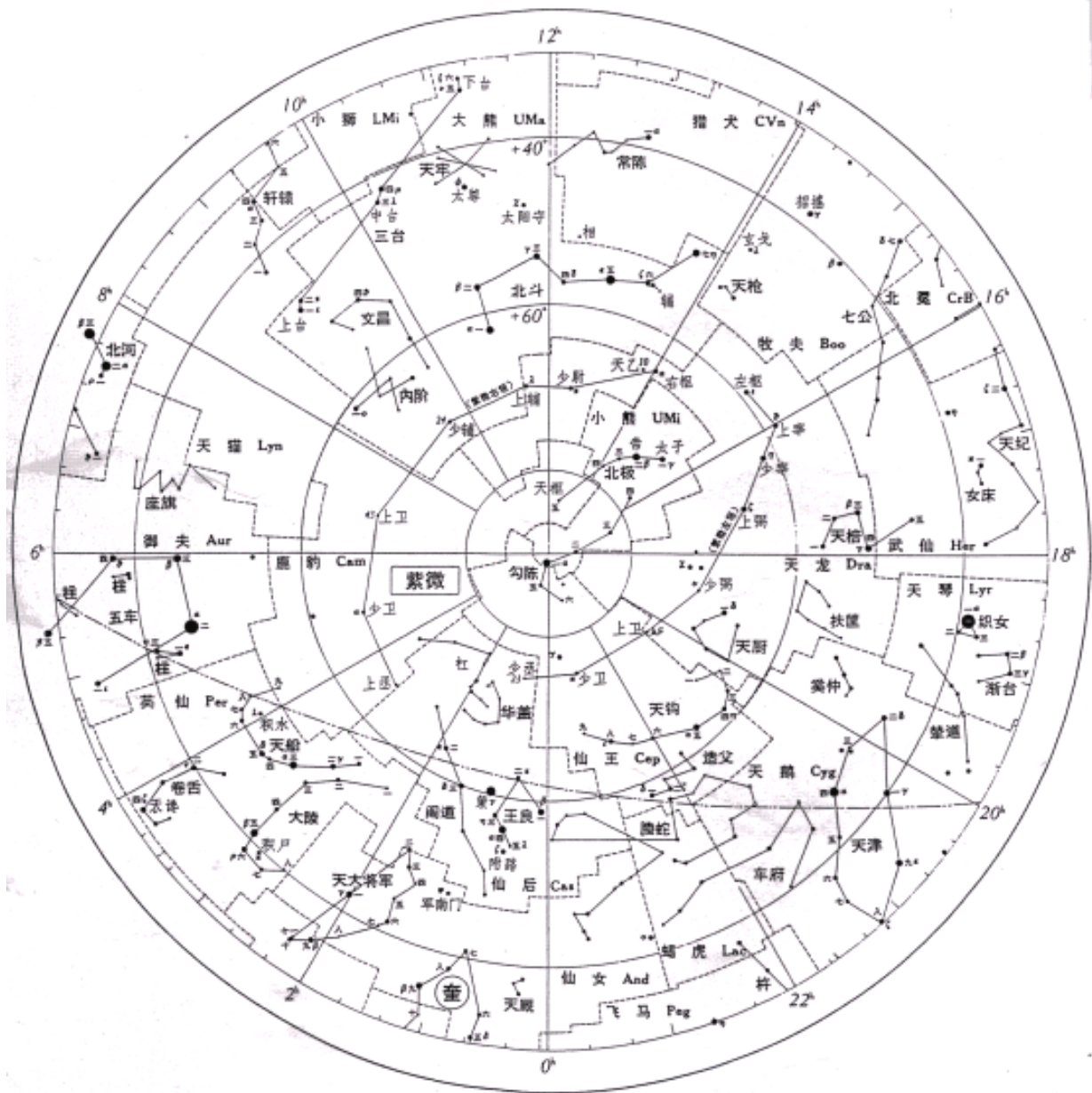


図 3: 伊世同氏による天文図、宋代 (960~1127年) 宋 (出典: [4])

### 3.2 古代中国の地上と天上

中国の星座はギリシャの星座と異なり、星の配列を具体的に表さないことが多い。天の北極近くの輝星を帝とし、近辺に大使・后・妾・庶子などの帝の一族、周辺に四輔・女史・女御・尚書など様々な役職や華蓋・調度品を置いている。その全体には侍臣と護衛官たちの星の列で囲み、紫微垣（帝の私的な居住宮殿）とする。これらは一年中地平線上に沈まない周極星の範囲にほぼ対応する。

現在のしし座・おとめ座・かみのけ座付近は太微垣（帝が執政するところ）の領域で、内五諸侯・九卿・三公などの主な官職が、文官・武官の星に囲まれている。へびつかい座・へび座・

ヘルクルス座付近は天市垣（天の市場）の領域である。市楼（市場管理者）・車肆（車を売る店）などが春秋戦国時代の国名のついた星の列に囲まれている。天の北極を中心に、紫微垣、その外の東西に太微垣・市微垣の三垣が並んでいる。それ以外の様々な星座は二十八宿による天の区分に従う。地上世界のあらゆるものが星座にされ、天文・気象や河川・地形、家族・伝説や歴史上の人物、施設、神や鬼なども星座にされている。便所や糞尿、その臭いを遮る屏風も星座にされ、天上も地上も同じ世界だと見ていたようだ。紀元前3世紀半ばに晋が三国を統一し、太史令の陳卓によって星座が整理統合され、1464星・283星座とした。黄道十二宮はインドに伝わり、隋・唐の頃にはシルクロード、南回りの海上ルートでも伝わっていった。明末・清初に西洋からきたキリスト教のイエズス会宣教師たちは、中国の各星座の観測の後、新たに星を追加した。中国から見えない星に関して、ヨーロッパの星座を紹介し、蜜蜂・十字架・金魚などの漢訳名を与えられ、この星座体系は清朝末まで用いられた。

### 3.3 日本へと伝わる星座（星宿）

日本には、古来すばるなどごく限られた星の名は存在したが、独自のまとまった体系はなく、中国の星座がそのまま導入された。キトラ古墳や高松塚古墳の天井に描かれたものは、朝鮮経由または直接伝わったとされる。

江戸時代になってから渋川春海は、1677年に版行した「天文分野之図」で、天の区分を日本の諸地方と対応させた独自の分野を公にした。1685年には初めて日本独自の暦を施行している。さらに中国の星座には属さず、これまで記載されてなかった308星について新たに位置を測定し、日本の制度になぞらえた独自の61星座を設定した。これを1698年に著した「天文瓊統」に記し、翌年「天文成象」という星図を公にした。これを図4に示す。



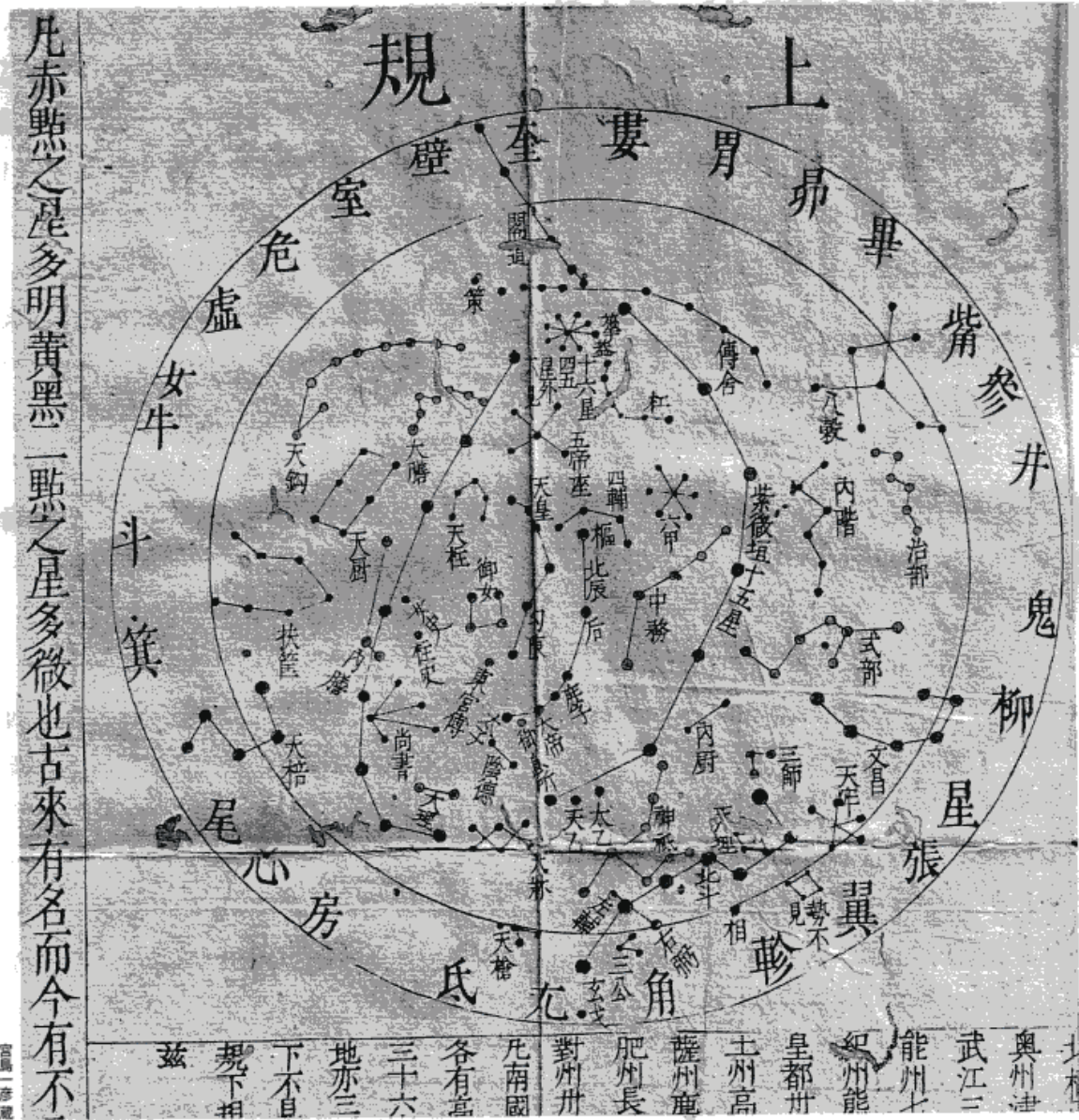


図 4: 天文成象、江戸時代 (1699 年) (出典: [2])

これらの和製星座は、それぞれ関連する星宿の近くに設けられた。星宿も日本の星座も近代国家が誕生して西洋文化が流入し、西洋の星座が世界の標準となり使われなくなった。



## 4 星宿図解説 - 高松塚古墳の天井画から

本章では高松塚古墳の壁画・天井画から、これらの画の意味や歴史などの概要を解説する。文献 [4] を参考にしている。

### 4.1 天井画・壁画の意味

昭和 47 年 3 月に高松塚を再発掘調査した際に天井画・壁画が発見された。石槨内の天井部に星宿、側壁面に日月、四神、人物群像が描かれていた。一般的な関心を集めたのは人物群像であったが、被葬者を含め検討する場合に、日月、四神、星宿の描かれている意義が重要となる。中国流では、天井に日月を描き、壁面に星宿を描く。しかし、高松塚では星宿を天井に描き、壁面に日月と四神が描かれているので墳墓の室内空間上、主要テーマは、日月と四神というふうに考えることが出来る。したがって、全て統一した思想表現であるといえる。壁面に描かれた四神というのは、本来は二十八宿を表すものである。高松塚古墳では天井画の星宿の方角に合わせて、壁面に四神が描かれている。つまり、壁面だけで二十八宿を現しているということがいえる。そして、日月はその二十八宿の上を通るので、その上の東西にある。高松塚で発見された壁画は、他国でも発見されている壁画と大差はない。しかし、天井画の星宿はかつてない発見と呼べる。図 5 に示す。



図 5: 高松塚古墳の壁画・天井画、694~710年、奈良明日香村（出典：[4]）

古代日本で紹介されている天文学は、中国の天文学である。この考えによれば、天は理想の国家であって、王、大臣、自衛などが存在し、理と制によって一大体系に組織されている。「空を仰ぐものの眼には厳として犯すべからざる権威を備へたるものとして教えられたのである。国を治める王侯や、学徳を極める聖人君子等は、かやうに系統化された天の理を知ることによつて、それぞれ己が道を行うたのであるが、斯くの如き天文学は所詮貴族的哲学体系であつて、一般衆俗の近づく可らざる厳格さを備えてゐた。」という見解がある。そこで、高松塚の星宿図の中国における思想的由来を分野説について検討してみる。分野説以外にも五帝説や六天説が存在している。

## 4.2 分野説について

まずは、分野という語義から説明する。「戦国の時、天文家が中国全土を天の二十八宿に配当して区別した称。其の分野に星変のある時は、其の国に災いがるといふ。」とある。つまり、分野説とは天上の星を地上の国や地域に配当し、それぞれに対応する天と地の間に密接不離の関係があるという説である。天の現象、星の変化を見ることで、この星を含む天の部分に相応する地域に起こる事象の吉凶を判断するために用いられ、古代中国の星占術の基礎となる理論である。しかし、中国古来において分野説は一定ではなかった。つまり、二十八の分野に分ける方法以外にも十二の分野に分ける方法も存在していた。

十二次とは、黄道に沿い天の一周を西から東へ十二等分したものである。それぞれに列国との関係を名称とともに与えられている。星紀は呉越、玄は齊、娵は衛、降婁は魯、大梁は趙、実沈は晉、鶉首は秦、鶉火は周、鶉尾は楚、寿星は鄭、大火は宋、析木は燕となり、後者が国の名前になっている。そして、これら前者の名前は星そのものの名称ではない。暦術でもこの十二次法と二十八宿法がある。晋書（648年）の天文志には、十二次の分野に二十八宿と十二支を配した魏の太史令陳卓という分類が上げられている。十二分野の法は木星と十二次との関係に基礎を置くもので木星の運行が約12年で天を一周するので十二次に配し、さらにこれらを地上の十二区域に配するものである。これを図6に示す。本説は後漢～魏にかけて確立し、唐に至って以降歴代これに準拠して変わることがなかった。この他にも十三分法や北斗七星を基準とする分野七分説も存在する。



ているものや、アプリケーション上に存在しない星が数多くある場合は星座線を描いていない。欠けている星が少ない場合は、飛ばして星座線を結んだり、他の星で代用している星座もある。これは現代の星座線を結ぶ際にも適応している。

古代中国で生まれた星宿と、古代アッシリアを起源とする現在の星座にはいくつか共通する部分がある。オリオン座を例に解説する。オリオン座は等級の高い星を四辺形とし、その中央に三列に輝く星を持つ星座である。星座線の引き方に違いはあるものの、どちらも共通する七つの星を持つ星座である。星宿では参宿と称されている。実際の開発画面より形を比較する。図7に示す。さそり座も比較的星が密集している箇所なので、房宿、心宿、尾宿という三つの星宿が密集している。明るい星や、特徴的な並びを持つ星などは地域が違ってても、同じような星座を結んでいる。詳しい解説や結果は第6章(40ページ)にて述べている。なお、本アプリケーションでは現代と古代の星座の対応付けは文献[8]を参考にしている。

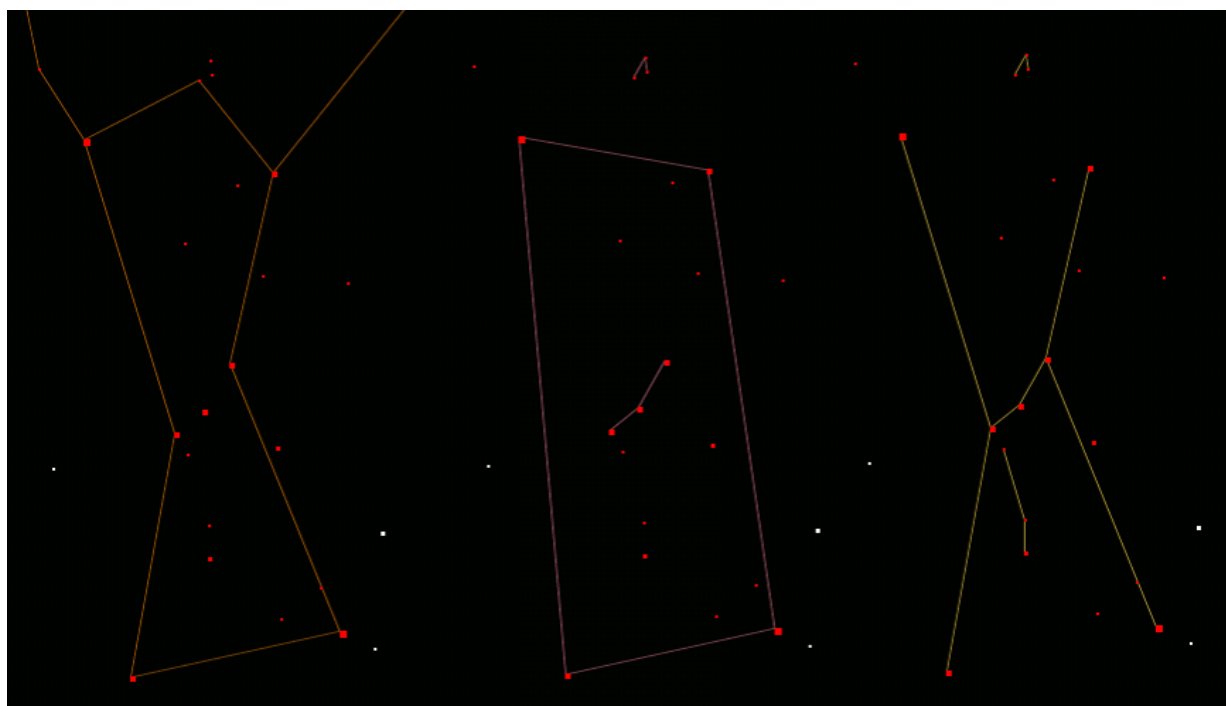


図 7: オリオン座の比較(左:オリオン座、中央・右:参宿)

## 5 星図アプリケーション解説

### 5.1 アプリケーションの概要

1001個からなる星をボタン操作で、三種類の星座線を描くことが出来るアプリケーションである。画面構成は、星空を描くディスプレイと、それを操作する右側に配置した11個のボタンからなる。アプリケーションの操作画面を図8に示す。

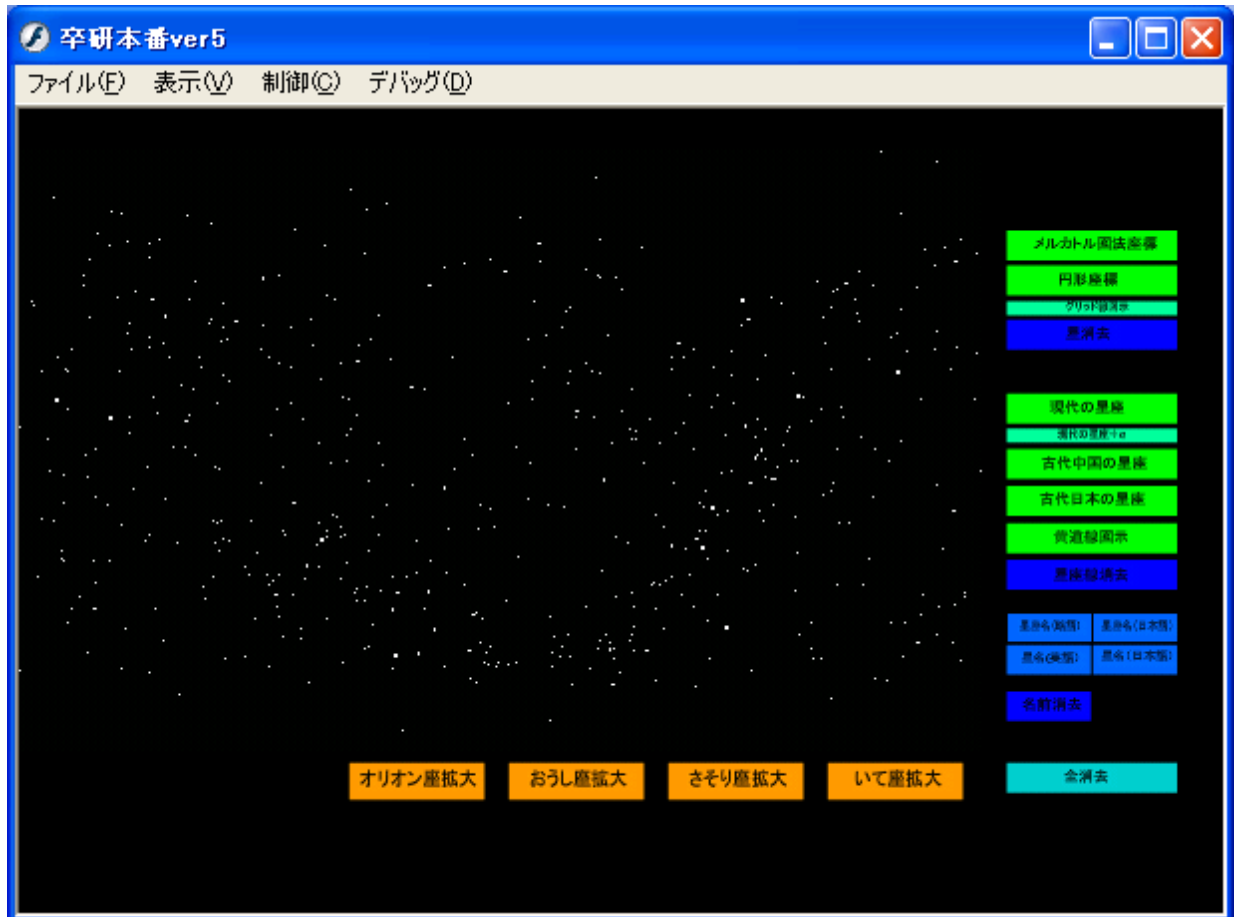


図 8: アプリケーションの操作画面

### 5.2 データベース構築

今回星座表に使用した星は文献 [6] と文献 [7] から引用している。文献 [6] のデータは、187 個の恒星の位置座標が載っており、当初はこのデータで画面作成をしたが、星の数が少なすぎたため、星座の形をほとんど取ることができなかった。文献 [6] からは、有名な恒星の名前 20 個のデータのみ参照している。187 の星で構成した最初のアプリケーション画面を図 9 示す。

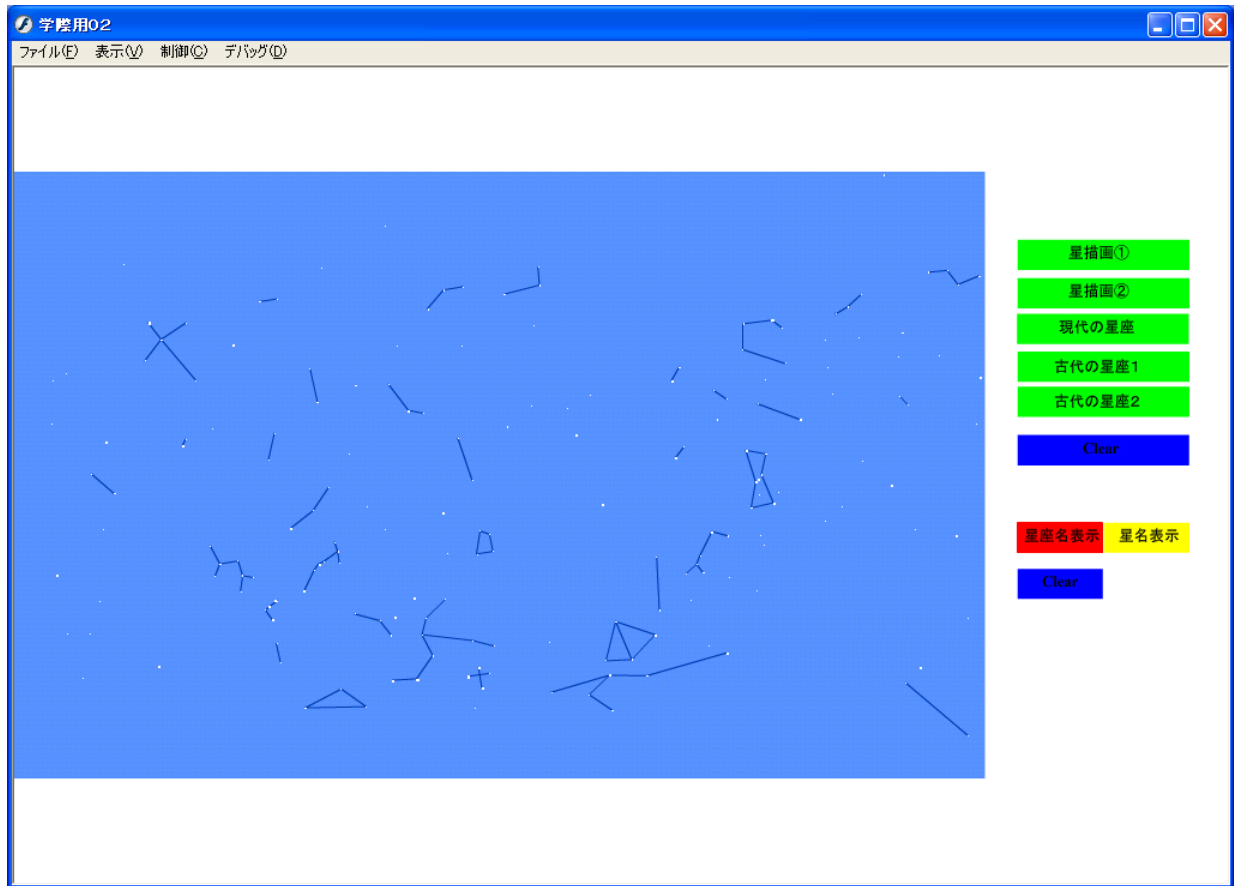


図 9: 187 の星で構成したアプリケーション画面

文献 [7] は、NASA の 9000 個の星のデータベースである。このデータベースは非常に詳細で、銀河座標系のデータなども含まれている。銀河座標は天球上の天体の位置を表す天球座標系の一種で、銀河中心と銀河面を基準とする座標系である。銀河座標では、天球上の緯度と経度として銀緯と銀経を使用する座標系である。このデータベースから等級、赤経、赤緯、各星に対応する星座の名前のデータを抽出する。赤経は時間・分・秒の 3 つのデータからなる 60 進法の値で、赤緯は度・分・秒の 3 つからなる赤経と同じく 60 進法の値である。各星に対応付けされた名前のデータは星座名の略名であるアルファベット 3 文字からなる値である。

本アプリケーションでデータベースとして使用した星は、明るい星順に - 1.46 ~ 4.6 等級の星で、1001 の星で構成している。1001 の星を使うに至った理由として、1000 個程星があれば、88 のほとんどの星座を表示することが出来るに十分な数であったことと、アプリケーションで使用したデータ数が 5024 個なのだが、Flash8 で 5000 を越えるデータを使用すると処理が出来なくなってしまう。そのため、処理が出来る限界のデータ数として赤経・赤緯・星座名略語・日本語星座名・英語星名・日本語星名の総数 5024 としている。

ActionScript のプログラム中では、赤経・赤緯のデータは文献 [7] から引用した形式では、3 つのデータに別れているので、時間・分・秒の 3 つを一まとまりにした 3 要素の配列を作り、それを 1001 要素の大きな配列に格納できるデータ形式に変換している。星の座標・星の名前・星座の名前は全て EXCEL を使用してデータを EXCEL 専用の関数を用いて加工した。以下に使用した全ての関数を示す。



- LEFT(文字列, 文字数) : 文字列の左端から指定した文字数だけ取り出す関数。
- RIGHT(文字列, 文字数) : 文字列の右端から指定した文字数だけ取り出す関数。
- MID (文字列, 開始位置, 文字数) : 文字列の指定した位置から指定した文字数だけ取り出す関数。
- & = CONCATENATE(文字列 1, 文字列 2 …, 文字列 N):文字列を結合する関数。

上の関数は、主に星の座標の度数や時間を 10 進数に変換するために用いた。次に、ある赤経データを例に説明する。00 08.4 というデータは 0 時 8 分 4 秒を表している。このデータを LEFT(文字列, 文字数) の関数を使い、時間の部分に相当する 00 を取り出す。次に MID (文字列, 開始位置, 文字数) を使い、分の部分に相当する 08 を取り出す。同様に、RIGHT(文字列, 文字数) を使い秒の 4 を取り出す。これら個別に取り出されたデータを配列として扱えるようにするために、CONCATENATE(文字列 1, 文字列 2 …, 文字列 N) を使い、それぞれのデータと [] を組み合わせ、[0,8,4] という形に加工する。

### 5.3 使用した古代・現代の星座

キトラ古墳や高松塚古墳などの天井画に描かれた星宿の中から二十八宿を表示可能にし、古代の星座は全域表示可能にした。しかし、星の数が少ないので、表現でき得る限りの星座・星宿だけを表示している。使用した星座を表 4 に、星宿は表 5 に示し、星座と星宿の対応関係を表 6 に示す。

表 4: 現代星座の表示の可・不可

星座名	略号	表示・可 / 不可
おおいぬ座	CMa	
りゅうこつ座	Car	×
うしかい座	Boo	
ケンタウルス座	Cen	
こと座	Lyr	
ぎょしゃ座	Aur	
オリオン座	Ori	
こいぬ座	CMi	
エリダヌス座	Eri	×
わし座	Aql	
おうし座	Tau	
さそり座	Sco	
おとめ座	Vir	
ふたご座	Gem	
みなみのうお座	PsA	×
みなみじゅうじ座	Cru	
はくちょう座	Cyg	
しし座	Leo	
つる座	Gru	
おおぐま座	UMa	
ほ座	Vel	
ペルセウス座	Per	
いて座	Sgr	
みなみのさんかく座	TrA	
くじゃく座	Pav	×
うみへび座	Hya	
おひつじ座	Ari	
こぐま座	UMi	
くじら座	Cet	
アンドロメダ座	And	
へびつかい座	Oph	
カシオペア座	Cas	
かんむり座	CrB	
りゅう座	Dra	
とも座	Pup	
おおかみ座	Lup	



ほうおう座	Phe	
ペガスス座	Peg	
ケフェウス座	Cep	
うさぎ座	Lep	
からす座	Crv	
てんびん座	Lib	
はと座	Col	
へび座	Ser	
はえ座	Mus	
ヘルクレス座	Her	
みずへび座	Hyi	
さいだん座	Ara	
きょしちょう座	Tuc	×
やぎ座	Cap	
りょうけん座	CVn	
みずがめ座	Aqr	
さんかく座	Tri	
インディアン座	Ind	
やまねこ座	Lyn	
コンパス座	Cir	
かじき座	Dor	
がが座	Pic	
レクチル座	Ret	
や座	Sge	
ぼうえんきょう座	Tel	
かに座	Cnc	×
コップ座	Crt	
うお座	Psc	
いるか座	Del	
らしんばん座	Pyx	
はちぶんぎ座	Oct	
とびうお座	Vol	
とかけ座	Lac	
こじし座	LMi	
ふうちょう座	Aps	×
たて座	Sct	
とけい座	Hor	×

ろ座	For	
こうま座	Equ	×
いっかくじゅう座	Mon	
じょうぎ座	Nor	
きりん座	Cam	
カメレオン座	Cha	
みなみのかんむり座	CrA	
ポンプ座	Ant	×
かみのけ座	Com	×
ちょうこくしつ座	Scl	
こぎつね座	Vul	×
ちょうこくぐ座	Cae	×
ろくぶんぎ座	Sex	×

表 5: 星宿の表示の可・不可

星宿名	高松塚古墳 表示・可/不可	トルファンの墳墓 表示・可/不可
角		
亢		
*		
房		
心		
尾		
箕		
室		
壁		
斗		
牛		
女		
虚		
危		×
奎		
婁		
胃		
昴		
畢		
觜		
参		
井		
鬼	×	×
柳	×	
星	×	
張	×	
翼		×
軫		

表 6: 星宿と星座の対応

星宿名	星座名
角	おとめ座
亢	おとめ座
*	てんびん座
房	さそり座
心	さそり座
尾	さそり座
箕	いて座
室	ペガサス座
壁	ペガサス座
斗	いて座
牛	やぎ座
女	みずがめ座
虚	みずがめ座
危	みずがめ座・ペガサス座
奎	アンドロメダ座・うお座
婁	おひつじ座
胃	おひつじ座
昂	おうし座
畢	おうし座
觜	オリオン座
参	オリオン座
井	ふたご座
鬼	かに座
柳	うみへび座
星	うみへび座
張	うみへび座
翼	コップ座・うみへび座
軫	からす座

\* ...氏の下に一。

#### 5.4 メルカトル座標

本アプリケーションは、天球上の星を世界地図でも使われているメルカトル図法と円形に北半球の星を描いた座標の二種類で表現している。この章ではメルカトル図法への変換に使用したプログラムから解説する。メルカトル図法では星の位置が極点に向かうにつれ歪んでいる。本来、星は天球上に座標を取っているのので、理科年表でも赤経・赤緯座標となっている。赤経は時間で、赤緯は度数で表示されている。赤経・赤緯座標を図 10 に、平面図の座標を図 11 に示す。

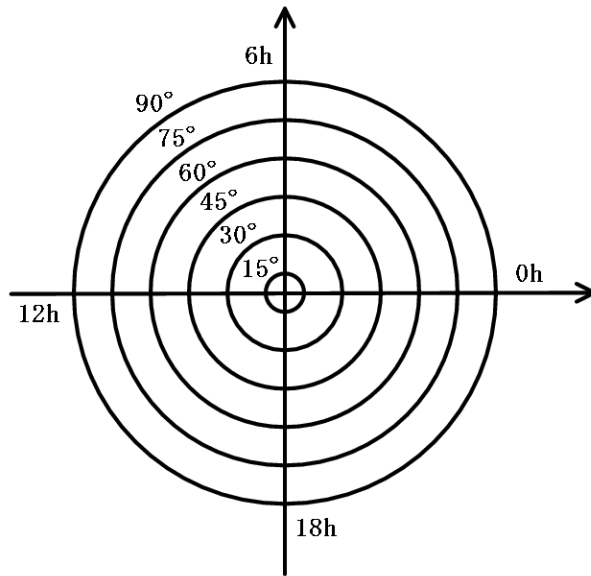


図 10: 赤経・赤緯座標 (北天のみ)

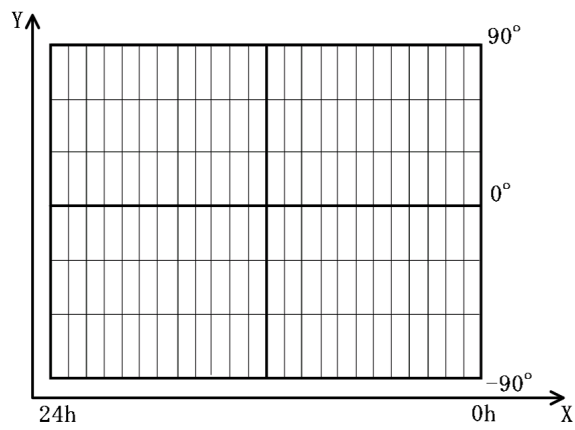


図 11: 平面図座標

図 11 のメルカトル図法は経度と緯度を格子状に配置させ、緯線と経線の比率を一定に保ったものである。直線上の方向は正確だが、距離や面積に関しては不正確と言う欠点がある。

メルカトル図法の平面図を描くために下記プログラム式で座標を変換している。

```
var A1 = preX[j];  
var A2 = preY[j];  
var B1 = preX[j] / 60;  
var B2 = preY[j] / 60;  
var C1 = preX[j] / 3600;  
var C2 = preY[j] / 3600;
```

preX[j]・preY[j] は、配列の配列から要素を取り出す配列である。配列の構成を下記に示す。

```
var xdata = [  
  [6,45,8.9],  
  [6,23,57.1],  
  [14,15,39.7],  
  [14,39,35.9],  
  [18,36,56.3],  
  [5,16,41.4],  
  .  
  .  
  .  
  ]
```

X 座標データ xdata の一部である。xdata が大きな配列で、その中の小さな配列に赤経データの時間・分・秒の 3 つのデータが格納されている。大きな配列の中の小さな配列から要素を取り出すために、要素数 3 の配列 preX[j]・preY[j] を定義している。A1,B1,C1 は赤経のデータで、A1 が時間、B1 が分、C1 が秒の変数で 60 進法から 10 進法にそれぞれ変換している。同様に、A2,B2,C2 は赤緯のデータで、A2 が度数、B2 が分、C2 が秒の変数で 60 進法から 10 進法に変換している。

次に、ActionScript の座標に合うように座標変換を行う。ActionScript の座標を図 12 に示す。

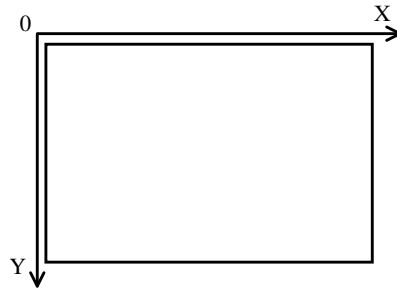


図 12: ActionScript 座標

```
var D1 = A1 + B1 + C1 ;
var D2 = A2 + B2 ;
```

D1 は赤経の 10 進法のデータの変数で、D2 は赤緯の 10 進法のデータとなっている。

```
x1 = D1*(Xsize/24);
y1 = D2*(Ysize/180)+(Ysize/2);
```

上記のプログラムの  $x1$  は、 $Xsize/24$  ( $Xsize$  には 1600 が格納されている) の部分で表示画面の X 座標が赤経と対応するように変換している。 $y1$  も同様に、 $Ysize/180$  ( $Ysize$  には 1000 が格納されている) の部分で表示画面の Y 座標が赤緯と対応するように変換されている。赤経は  $-90 \sim +90$  の範囲で座標を取るのので、 $+(Ysize/2)$  の部分で修正を加えている。

```
theX = Xsize - x1;
theY = Ysize - y1;
```

最後に、 $theX$  は天球上では通常の座標とは違って左右逆に進む座標を取るのので表示画面の X 座標のサイズから引いている。 $theY$  も同様に、こちらは ActionScript の座標が下降方向に正を取るのので表示画面の Y 座標のサイズから引いている。

## 5.5 円座標

この章では円形の座標変換について説明する。今回開発したアプリケーションでは、北半球を北極星中心に見た星表にするためにこの座標変換を行った。ちょうど図 10 のように座標を取る。

```
var Rmax = 90;
var R = (Rmax-dosuuY) / Rmax ;
var h = 6 ;
var radX = Math.cos( (zikanX * 360 / 24 * Math.PI * 2 / 360) +
    (h * 360 / 24 * Math.PI * 2 / 360) ) *R;
return(radX);
```

上のプログラムはサブルーチンとして使用している。赤経、赤緯ともに角度を表しているものなので、メルカトル図法と同様に、一度 10 進数に変換する。zikanX は赤経の 10 進数を読み込む変数で、dosuuY は赤緯の 10 進数を読み込む変数である。まず、dosuuY を角度のデータであるが半径と見立てて使用する。Rmax は赤緯が北半球は 0 ~ 90 度なので、最大値である 90 を格納している。変数 R は角度であるデータの比を取り、半径として扱えるように計算している。変数 radX は星の x 座標を取り出している。式 (1) に示す。

$$x = \cos \theta \times R \quad (1)$$

この式を使用し、radX に x 座標が与えられている。zikanX は 1 ~ 24 の値を取る変数なので、一般的に扱える度数に変換するために 360 / 24 をかけている。度数に変換後に、ActionScript の Math.cos() の関数に適応させるために radian に変換しなければならないので、Math.PI \* 2 / 360 をかけて変換している。cos の中で (h \* 360 / 24 \* Math.PI \* 2 / 360) を足しているが、これは 90 度を足しているプログラムである。これにより、座標の始まりの位置を変更している。y 座標も同様に、sin を使用して y 座標を取り出している。

## 5.6 各操作ボタン

本アプリケーションを操作する上で、主軸になってくるのがボタンである。本章では各ボタンの目的・働き、プログラムの仕組みなどを解説する。

### 5.6.1 星描画ボタンと機能の解説・工夫

本アプリケーションでは、星描画ボタンが 2 種類あり、星を消去するボタンと合わせ 3 種類のボタンがある。星を消去するボタンは、星座線のみを見たいという場合を考えて作った。図 13 に示す。

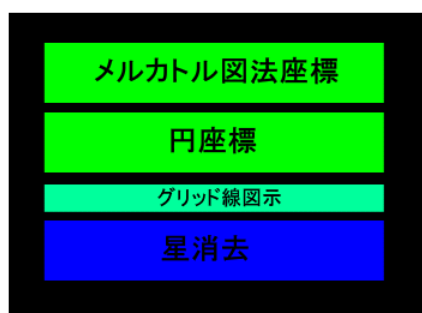


図 13: 描画ボタン・グリッド描画ボタン・消去ボタン

メルカトル図法座標のボタンがメルカトル図法の座標変換で、円座標のボタンは円形の座標の変換に対応している。座標変換されたデータを元に星を描くプログラムはサブルーチン化し



ている。グリッド線図示ボタンはX座標を半分にする線とY座標を半分にする線の2本だけ描画する。X座標のデータとY座標のデータと等級のデータで、星に見立てた正方形を描く。円ではなく正方形を選んだのは、角を星の輝きに見せるためである。

星を消去するボタンは、画面上に背景と同じ色の長方形を描いて見かけ上、星を消しているように見せている。

### 5.6.2 星座線ボタン

星座線描画ボタンは4種類あり、星座線消去ボタンと黄道線図示ボタンを合わせて6種類ある。図14に示す。

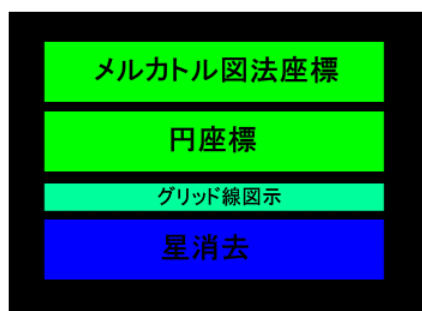


図 14: 星座線描画ボタン・黄道線図示ボタン・消去ボタン

現代の星座ボタンは、現在一般的に知られている星座を描く。現代の星座+ ボタンは、春の大三角・夏の大三角・冬の大三角・冬のダイヤモンドの4種類を描画する。古代中国の星座(唐代)ボタンはトルファンのアスターナ古墳群で発掘された古代中国の星座を描く。古代日本の星座(飛鳥時代)ボタンは高松塚古墳で発掘された古代日本の星座を描く。星座線を描くプログラムもサブルーチン化している。現代の星座名は88個、古代中国・日本の星座名は共に28個となっている。黄道線図示ボタンは太陽の通り道である黄道をメルカトル座標上と円座標上に描く。ベジエ曲線で黄道を近似している。

### 5.6.3 名前表示ボタン

星座名(略語)・星座名(日本語)の2つのボタンと星名(英語)・星名(日本語)の2つのボタン、合わせて4種類と、消去ボタンの合わせて5つのボタンを用意している。図15に示す。

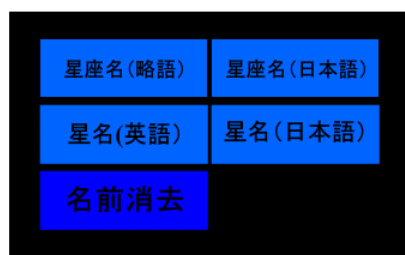


図 15: 名前表示ボタン・消去ボタン

星座名は、X座標・Y座標データと同様に、各星毎に星座名を配列データとして持っている。その中から同じものを選び取り、新たな配列に入れなおし、星座名表示ボタンで表示させている。抽出方法や、プログラムの工夫は次章で解説する。星名は各星の座標データを持つ配列として定義している。本アプリケーションでは、名前を持つ星を文献[6]より引用している。名前を持っている星は20個となっている。

#### 5.6.4 特定の星座拡大ボタン

現代の星座と古代中国・日本の星座を比較する中で、注目して欲しい特定の4星座だけを拡大表示するボタンである。オリオン座、おうし座、さそり座、いて座の4種類用意している。例として、オリオン座だけ拡大するボタンを図16に示す。



図 16: オリオン座だけ拡大ボタン

この外に、星・星座線・各々の名前も単独で全て消去する全消去ボタンも備えている。

## 5.7 星座名の抽出・配置・表示

1001の星のデータは各々星座名のデータを持っている。星座名のデータを持たない星も数個であるが存在する。同じ星座名を持つデータを一くくりにし、1001個データから88個の星座名だけを抽出する。条件判定し、同じ名前を省きながら新たな配列に星座名を代入する。この抽出する部分はサブルーチン化している。

次に、星座名を配置する方法を解説する。まず、星座名を表示するためのテキストフィールドを生成する。生成後の配置場所の決定方法は、次に示す。

```
if (NaCou[i] == starName[j]){  
Xsum = Xsum+gX[j];  
Ysum = Ysum+gY[j];  
k++;  
}
```

このプログラムは二重ループの中で使われている。配列 NaCou[i] は、抽出された88個の星座名データを持つ。starName[j] は各星の星座名データの配列である。gX[j]、gY[j] は各星のX座標・Y座標データである。その座標データをX座標はXsumに、Y座標はYsumに足しこまれる。変数kは平均値を出すときに使う全体の個数を表す。

```
Xave = Xsum/k;  
Yave = Ysum/k;
```

上のプログラムのXave、Yaveは平均値を格納する変数である。この座標データの位置にテキストフィールドが生成される。

```
createTextField("WNA"+i, i+1, Xave, Yave, 30, 20);  
this["WNA"+i].multiline = true;
```

createTextField関数は、テキストフィールドを生成する関数である。""で囲まれているのがテキストフィールドの名前である。iはテキストフィールドの番号で、Xave、Yaveの位置に生成し、30pixがテキストフィールドの幅で、20pixが高さとなっている。this["WNA"+i].multiline = trueの部分で複数行の文字データも表示出来るように定義している。

Flash8では、テキストフィールドが生成と同時に文字データを出力することが出来ない。そのため、テキストフィールドの数だけ文字データを入力する関数を定義しなければならない。

```
WNA0.text=NaCou[0];
```

WNA0.textはテキストフィールド自体を指している。0番～名前数だけこの関数を用意する。この関数もEXCELを使い、専用の関数で個数分用意する。複数同じ関数を定義しなければならない時がFlash8を使っているとよく起こる。たびたびEXCELを使いながらデータの加工を行っている。

## 5.8 拡大表示機能

画面上の好きな場所をクリックすると拡大することが出来る。クリックした座標から X 座標は右方向に画面サイズの 1/4、Y 座標も下降方向に画面サイズ 1/4 の範囲の星を 4 倍拡大で表示することが出来る。

```
moveX = _root._xmouse;  
moveY = _root._ymouse;
```

`_root._xmouse` と `_root._ymouse` の関数でクリックした位置の X 座標・Y 座標のデータを取得することが出来る。このデータを `moveX`、`moveY` に格納する。

```
wakuX = moveX + ( Xsize/4 );  
wakuY = moveY + ( Ysize/4 );
```

`moveX` に画面サイズの 1/4 を足す。Xsize/4 が X 座標の最大値の 1/4 を表している。Y 座標も同様に計算する。`wakuX` と `wakuY` に格納する。

```
upX1 = (gX[i] - moveX) * 4 ;  
upY1 = (gY[i] - moveY) * 4 ;
```

条件判定で、`moveX ~ wakuX` と `moveY ~ wakuY` の間の領域が拡大する対象領域となる。`gX[i]`、`gY[i]` は拡大前の元の座標データで、クリックした箇所の座標データを引いておくことで元の座標に対応させている。X・Y 共に 4 倍し、`upX1`、`upY1` に格納し、任意の箇所の拡大となる。拡大機能の略図を図 17 に示す。

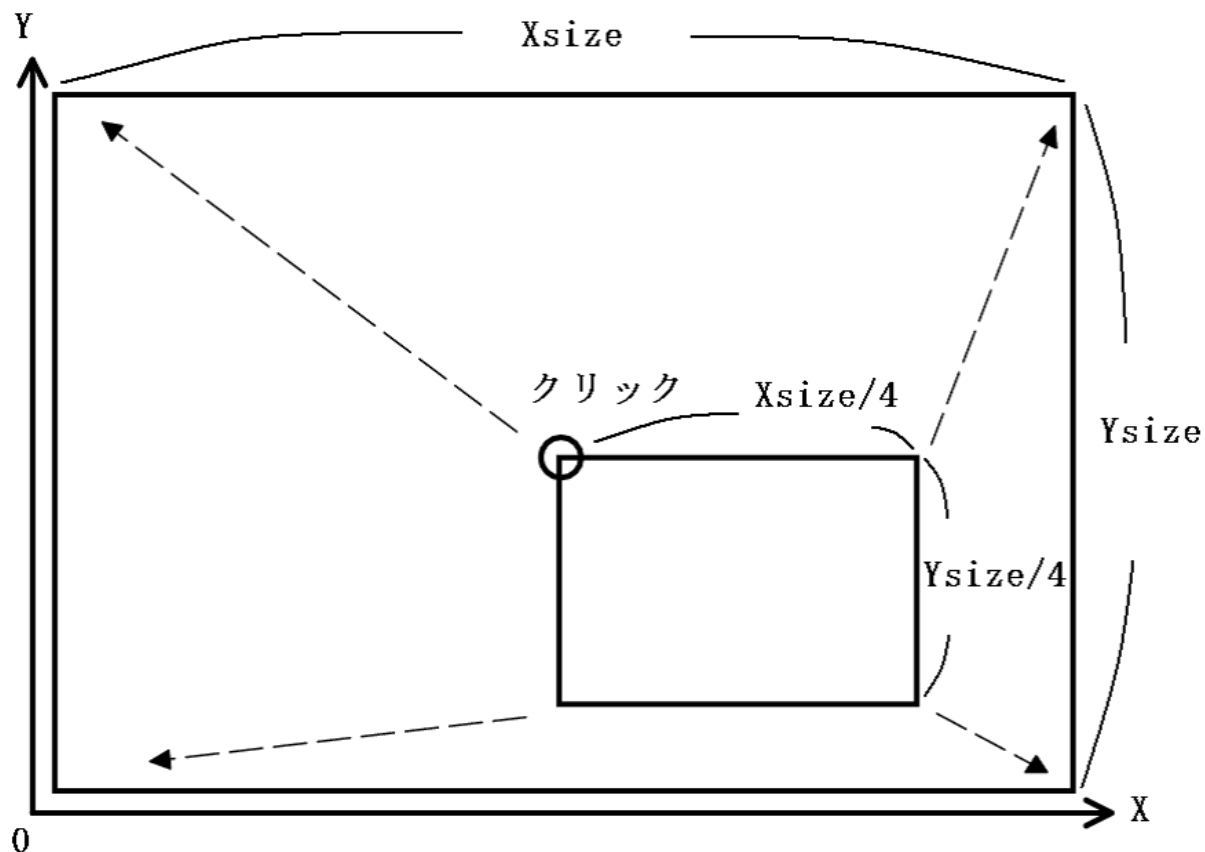


図 17: クリック拡大

## 5.9 星座線描画機能の解説

星座線を同定したり、拡大や円形座標に対応するように行ったプログラム上の工夫について本章で解説する。

### 5.9.1 星の同定方法

ボタンを押すことで星座線を描くことが出来るが、各星の対応付けを行う必要がある。点と点を結ぶ作業で行った工夫などについて解説する。うさぎ座を例にし、対応付けの作業を図 18 に示す。

予め、全ての星数のテキストフィールドとそこに描く文字を定義しておく。次に、拡大したときに同時に必要なテキストフィールドを星に対応するように配置する。テスト用のボタンを作っておき、押すことで必要な番号だけが対応する座標に表示される。

もう一つ行った工夫は、うさぎ座の名前を持つ座標を選び取り、赤色で点を描くように条件判定を付けたことである。実際には線を結ばないが、その星座の星として見られている星が存在する。図 18 で例をとると、844 番の星がそうである。そのために、色を変える作業をしなければ線を結ぶための星を同定することが困難であった。色を変える作業と番号付けをすることで、作業効率は大幅に上がった。

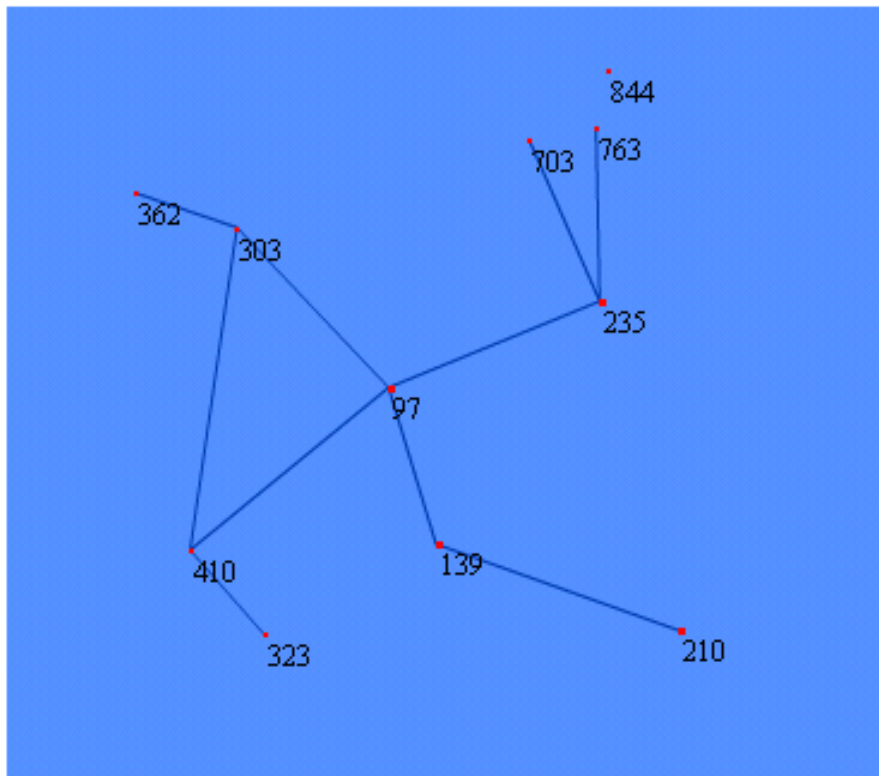


図 18: 点と点の対応付け (開発中画面より)

### 5.9.2 座標ごとの対応

本アプリケーションでは、様々な座標変換を行っている。星座線を描く場合にそれぞれの対応を考える必要があった。

```
_global.JUD = [ (xdata.length) ];
_global.Judge = 0;
```

JUD は全ての星数分の要素を持つ配列で、0~2のどれかのデータを持っている。変数 Judge は座標変換されるたびに0~2の数字が格納される。メルカトル図法の座標変換は0、拡大座標変換では1、円形座標変換では2が格納される。配列 JUD も座標変換の際に、表示に必要な座標が選択される度に、0~2のどれかのデータが格納される。変数 Judge と配列 JUD の値が一致するデータだけ選ばれて、描画する座標となる。

## 5.10 Flash8でのデータの取り扱い

本アプリケーション開発で、Flash8のデメリットを発見した。データベース構築時に、データの加工上0から始まるデータをいくつか作った。ただの0のデータなら扱うことが出来るのだが、01や02といった数字を扱うことが出来ない。元のデータが0から始まる表記であった

ので、全てのデータを修正する必要があった。データ数も今回は星の数だけで1001個のデータを使っているが、本来は3000の星を使う予定であった。Flash8の機能上、総数で5000以上のデータを扱うことが出来ない。大量のデータが必要となる計算はFlash8では向いていないということが分かった。

## 6 比較による結果

本アプリケーションの開発で分かったことは、現在の星座で星座占いや誕生日に対応した星座などは、黄道 12 星座と呼ばれている。黄道すなわち太陽の通り道に来る星座のことを指している。図 19 からそれが再認識された。図 19 上の  $\sin$  曲線が、天球上の黄道となる。古代中国・日本の星座も、星宿の中の二十八宿は太陽の宿る星宿という意味がある。古代の二十八宿も現代の星座と同じように、黄道に沿って配置されている。図 20、図 21 より、黄道に沿って星宿が配置されていることが改めて確認することが出来た。

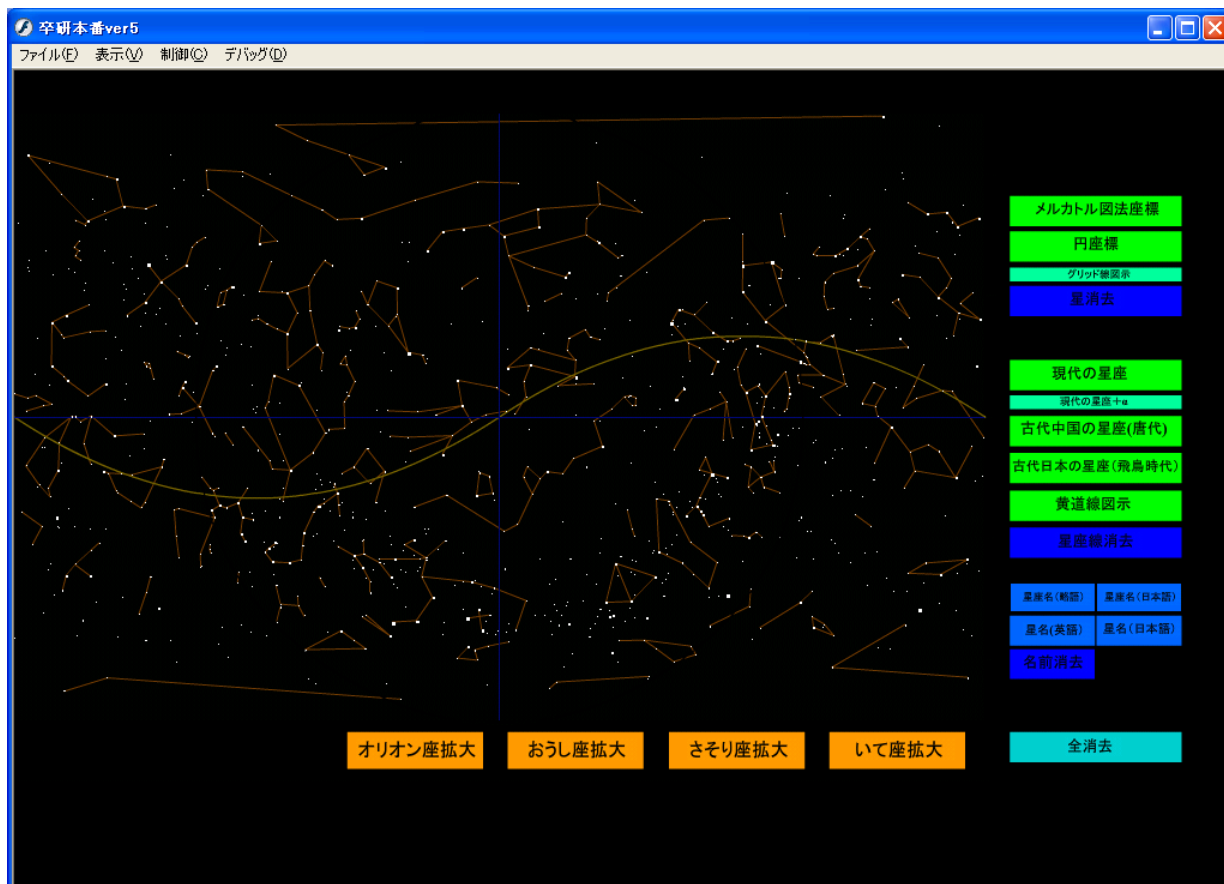


図 19: 現代の星座と黄道 (88 星座)



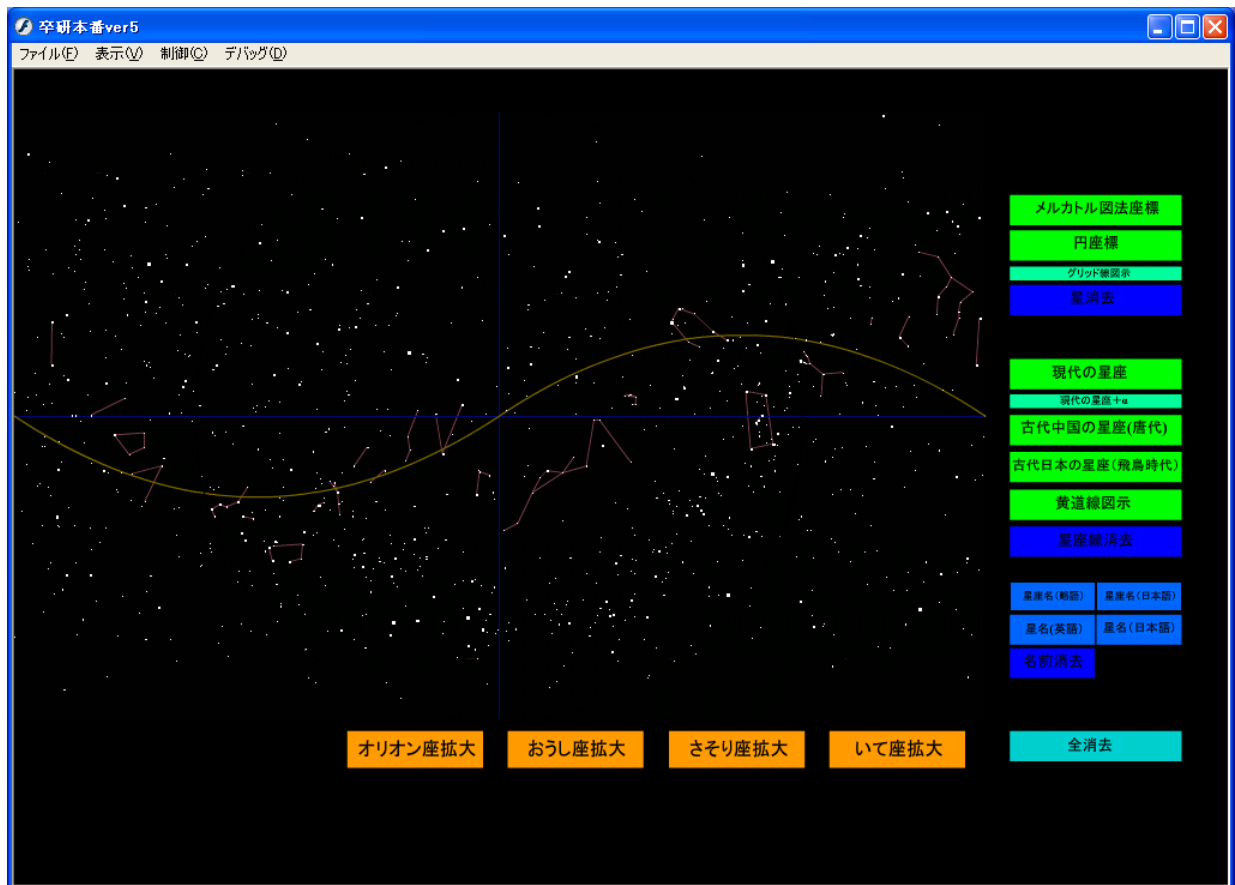


図 20: 古代中国の星宿と黄道線

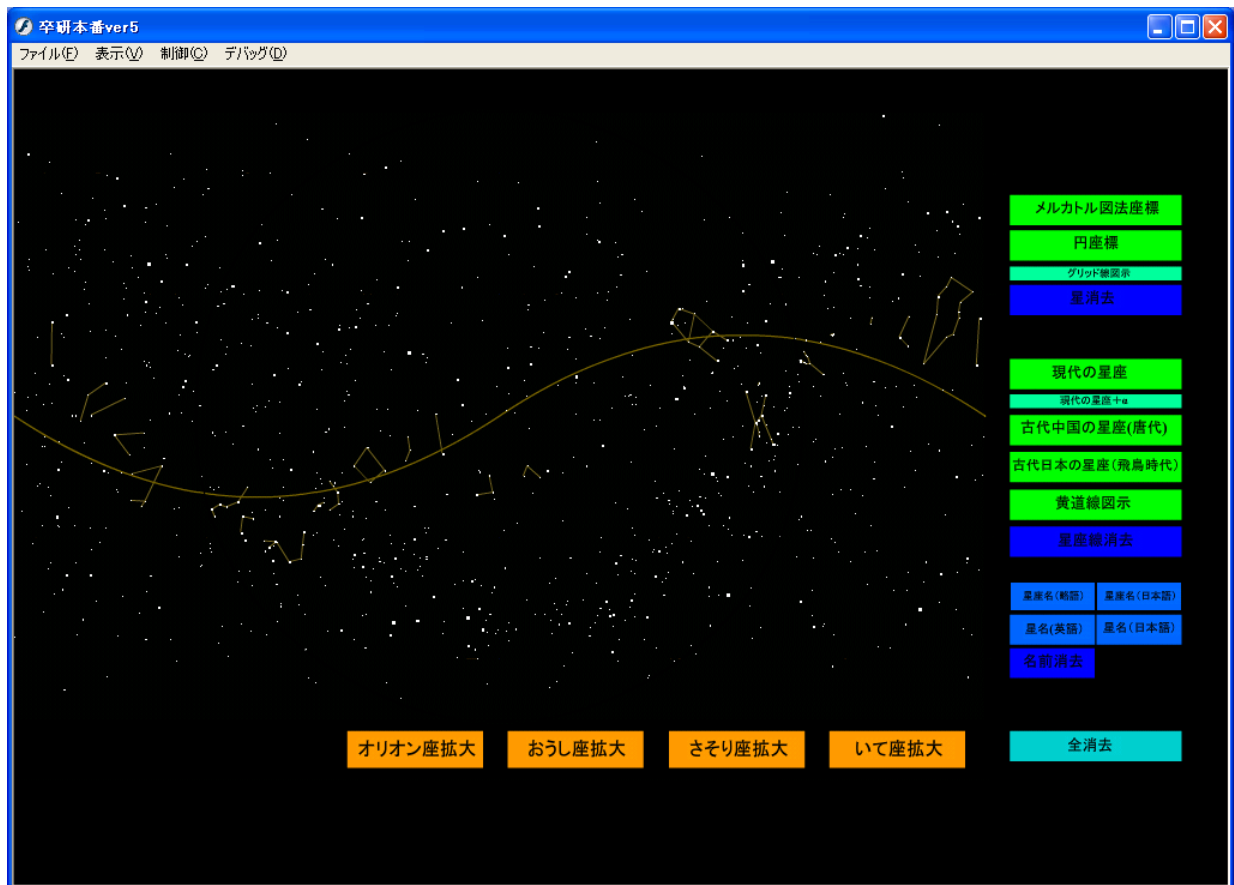


図 21: 古代日本の星宿と黄道線

作られた場所や観測された場所が違って、明るい星・特徴的な色をした星は必ず星座として使われている。さそり座も現代・古代共に星座を象られているものの一つである。さそり座に含まれる最も特長な星が赤色で1.09等のアンタレスである。アンタレスの左右両隣にほぼ等間隔で並ぶ星がある。3つ並んだ星が印象的で、二十八宿ではその3つの並びが単独で心宿という星宿を形成している。前章で紹介したオリオン座と同様に、さそり座も古代中国・日本の星宿に酷似した形となっている。図22、図23、図24にそれぞれ示す。

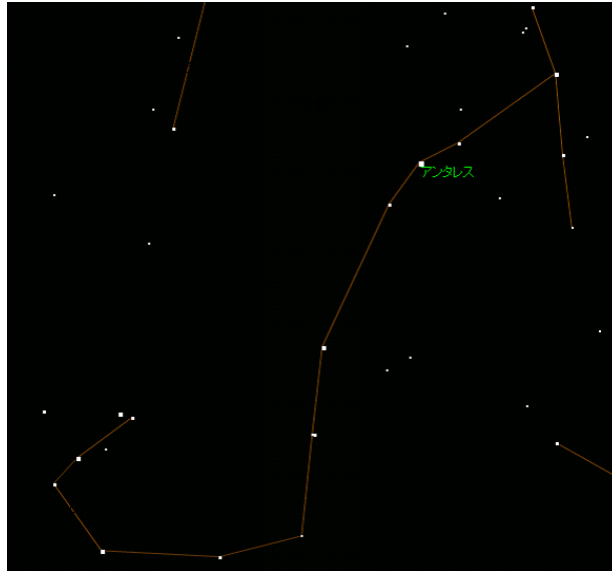


図 22: さそり座の拡大図

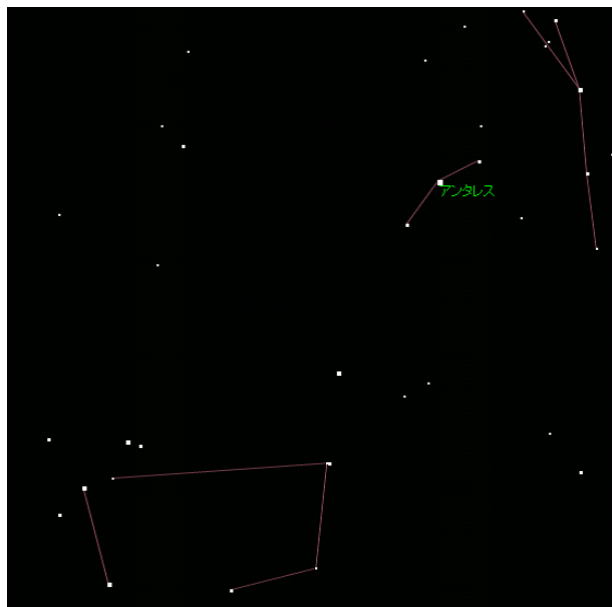


図 23: 古代中国の星宿（左：尾宿 中央：心宿 右：房宿）

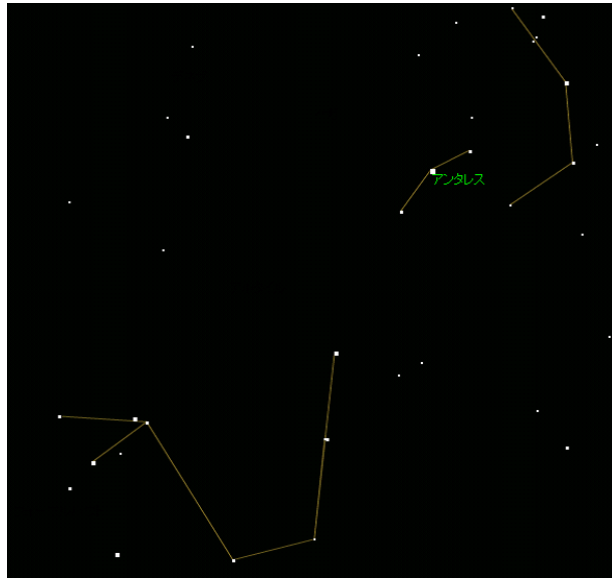


図 24: 古代日本の星宿 (左: 尾宿 中央: 心宿 右: 房宿)

次に、同じ星を使っているが、違う形に結ばれている星座も存在する。いて座は大きな半人半馬が弓を構えている姿に見立てられるが、二十八宿では小さな星宿が象られている。いて座に相当する部分に2つあり、箕宿・斗宿とそれぞれ呼ばれている。日本では斗宿のことを北斗七星に対して南斗六星と呼んでいる。いて座の中心部分の明るい星が北斗七星のような形に並んでいるためである。図 25、図 26、図 27 にそれぞれ示す。

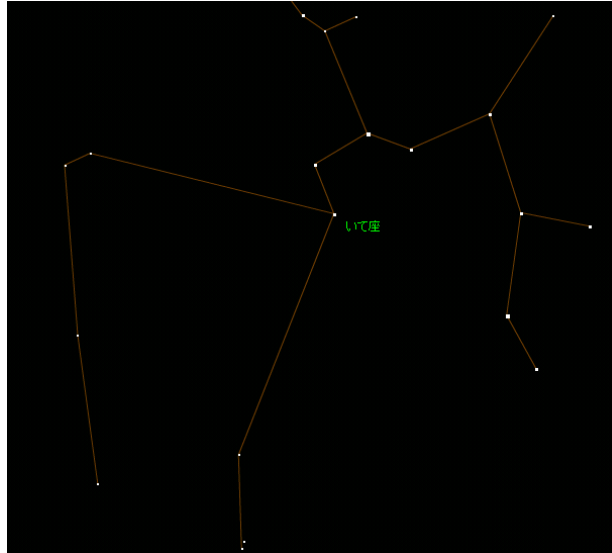


図 25: いて座の拡大図

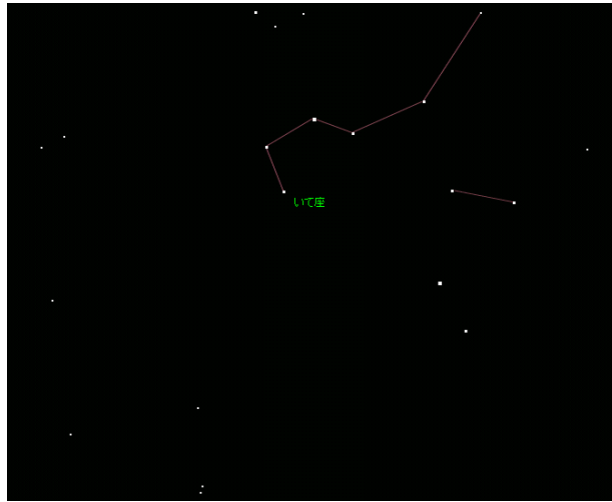


図 26: 古代中国の星宿（左：斗宿 右：箕宿）

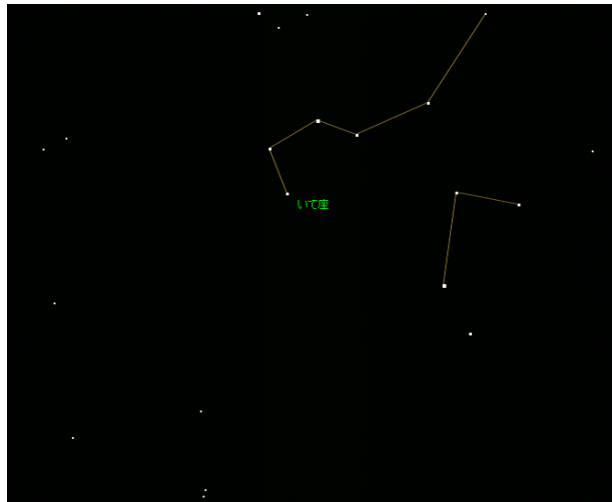


図 27: 古代日本の星宿（左：斗宿 右：箕宿）

本研究で再確認されたことは、異文化で発達した異文明であっても見上げている星は同じであるということである。星の捉え方や星座の形の結び方が違ってても、星の並びの特徴的な部分は共通する部分が多い。また違った形を象ろうとも、その星の特徴的な並びを生かしているのので、全く違った形を作るものは少なかった。

## 7 まとめ

今回、現代の星座と古代の星座を対応させる星図アプリケーションの開発で、星座についての歴史や当時の人々の思いを感じることが出来た。日本人は自国の文化を大切にしない民族である。そのため、このアプリケーションに触れて文化を感じて欲しい。日本の文化に興味を持ってもらえるきっかけになり得たら幸いである。現在当たり前に見ている星座にも、長い歴史や変遷がある。星座に付随する神話もただの御伽噺ではなくて、当時の政治的な背景や、隣国との関係性などにも深く関連している。自国の文化に触れ、他国の文化にも興味を持って欲しい。私自身が勉強する中で、そこに存在するものにはかならず歴史があるということを知って欲しいと考え、そこに重きを置いてアプリケーションの開発をすることが出来た。

Flash8 はインタラクティブなアプリケーションを作ることが出来るが、大規模な計算には向かないことが分かった。分かりやすく、使いやすいアプリケーションを作るという意味では優れたツールである。今回はターゲットの年齢を低く設定していたので、本研究では最適なツールであったといえる。

以上のことから、手軽に天文学に触れることができるアプリケーションが作ることが出来た。Web で公開し、世界中の人にも使っていただけたら幸いである。

## 8 参考資料

### 参考文献

- [1] 日本天文学会編,「世界星座早見」,三省堂(2003)
- [2] B.E. シェーファー,「日経サイエンス」,日経サイエンス社(2007.2)
- [3] 有坂隆道,「古代史を解く鍵 暦と高松塚古墳」,講談社(1999.2.10)
- [4] 綱干善教,「橿原考古学研究所論集第六 古代における星辰図について - 高松塚の星辰図を中心に - 」,(1984)
- [5] 飛鳥古京顕彰会,「新改訂 キトラ古墳と壁画」,明日香村観光開発公社(2006.3.27)
- [6] 国立天文台,「理科年表 CD-ROM2001」,丸善株式会社(200.11.30)
- [7] BSC5P - Bright Star Catalog(NASA(<http://heasarc.gsfc.nasa.gov/W3Browse/star-catalog/bsc5p.html>))
- [8] 古代中国の星座たち～黄道二十八宿を星空にたどる (Stellar Scenes(<http://www.ne.jp/asahi/stellar/scenes/index.html>))

## 9 付録

現在星座占いなどで使われている星座を古代の星座で対応付けしたものを付録として表7に示す。最後にプログラムソースも添付している。

表 7: 星宿占い表

誕生日	星宿名	読み	星座名
8/23 ~ 9/7	角宿	かくしゆく	おとめ座
9/8 ~ 9/22	亢宿	こうしゆく	おとめ座
9/23 ~ 10/23	* 宿	ていしゆく	てんびん座
10/24 ~ 11/1	房宿	ぼうしゆく	さそり座
11/2 ~ 11/11	心宿	しんしゆく	さそり座
11/12 ~ 11/21	尾宿	びしゆく	さそり座
11/22 ~ 12/7	箕宿	きしゆく	いて座
12/8 ~ 12/21	斗宿	としゆく	いて座
12/22 ~ 1/19	牛宿	ぎゅうしゆく	やぎ座
1/20 ~ 1/29	女宿	じょしゆく	みずがめ座
1/30 ~ 2/8	虚宿	きょしゆく	みずがめ座
2/9 ~ 2/18	危宿	きしゆく	みずがめ座・ペガスス座
2/19 ~ 2/28	室宿	しっしゆく	ペガスス座
3/1 ~ 3/10	壁宿	へきしゆく	ペガスス座
3/11 ~ 3/20	奎宿	けいしゆく	アンドロメダ座・うお座
3/21 ~ 4/4	婁宿	ろうしゆく	おひつじ座
4/5 ~ 4/19	胃宿	いしゆく	おひつじ座
4/20 ~ 5/4	昴宿	ぼうしゆく	おうし座
5/5 ~ 5/20	畢宿	ひっしゆく	おうし座
5/21 ~ 6/5	觜宿	ししゆく	オリオン座
6/6 ~ 6/21	参宿	しんしゆく	オリオン座
6/22 ~ 6/30	井宿	せいしゆく	ふたご座
7/1 ~ 7/8	鬼宿	きしゆく	かに座
7/9 ~ 7/17	柳宿	りゅうしゆく	うみへび座
7/18 ~ 7/26	星宿	せいしゆく	うみへび座
7/27 ~ 8/4	張宿	ちょうしゆく	うみへび座
8/5 ~ 8/14	翼宿	よくしゆく	コップ座・うみへび座
8/15 ~ 8/22	軫宿	しんしゆく	からす座

\* ...氏の下に一。



## メインプログラム

```
import flash.display.BitmapData;
import flash.geom.Rectangle;
_global.Ysize = 1000;
_global.Xsize = 1.6 * Ysize;
this._bmd = new BitmapData(Xsize,Ysize,true,0x88000000);//0xAA0055FF
this._mc=this.createEmptyMovieClip("bm_mc",this.getNextHighestDepth());
this._mc._x=0;
this._mc._y=0;
this._mc.attachBitmap(this._bmd,this.getNextHighestDepth());
this.StarPix();
this.seizaLine();
this.GLine();
this.NameCount();
this.StarDraw();
this.ClearDraw();
this.xDrawCircle();
this.radGetX();
this.radGetY();
```

```
function StarPix() {
```

```
%%%ここに星のデータベースが入る。%%%
```

```
//大域 X , Y , COLOR , Class
_global.MerX = [ (xdata.length) ];
_global.MerY = [ (ydata.length) ];

_global.fX = [ (xdata.length) ];
_global.fY = [ (ydata.length) ];

_global.gX = [ (xdata.length) ];
_global.gY = [ (ydata.length) ];

_global.ClassDef = [ (Class.length) ];
```

```
//線の色
_global.COL1 = 0xFFEF810F;

//線消す色
_global.COL2 = 0xFF000000;

//黄道線の色
_global.COL3 = 0xFFFFD400;

//グリッドの色
_global.COL4 = 0xFF00FF0;

//中国色
_global.COL5 = 0xFFFF87A0;

//日本色
_global.COL6 = 0xFFAFDFE4;

//大星座色
_global.COL7 = 0xFFFF0000;

//星の色
_global.StarC1 = 0xFFFFFFFF;

//星消す色
_global.StarC2 = 0x88000000;

_global.JUD = [ (xdata.length) ];
_global.Judge = 0;

_global.HsizeX = Xsize/2;
_global.HsizeY = Ysize/2;

//比較判定用
_global.DrawJ1 = 0;
_global.DrawJ2 = 0;
_global.DrawJ3 = 0;
_global.DrawJ4 = 0;
_global.DrawJ5 = 0;
```

```

// 星描画
for (var i = 0 ; i < xdata.length ; i ++ ) {
var preX = xdata[i];
var preY = ydata[i];
for (var j = 0 ; j < preX.length ; j ++ ) {
if (j==0) {
var A1 = preX[j];
var A2 = preY[j];
} else if (j==1) {
var B1 = preX[j] / 60;
var B2 = preY[j] / 60;
if (A2 < 0) {
B2 = B2 * (-1);
}
} else {
var C1 = preX[j] / 3600;
var C2 = preY[j] / 3600;
}
}
var D1 = A1 + B1 + C1 ;
var D2 = A2 + B2 + C2;

fX[i] = D1;
fY[i] = D2;

x1 = D1*(Xsize/24);
y1 = D2*(Ysize/180)+(Ysize/2);

theX = Xsize - x1;
theY = Ysize - y1;

if (-10 < Class[i] && Class[i] < 0) {
ClassDef[i] = 6;
} else if (0 <= Class[i] && Class[i] < 1.5) {
ClassDef[i] = 5;
} else if (1.5 <= Class[i] && Class[i] < 2.5) {
ClassDef[i] = 4;
} else if (2.5 <= Class[i] && Class[i] < 3.5) {
ClassDef[i] = 3;
} else {
ClassDef[i] = 2;
}

```

```

}

MerX[i] = theX;
MerY[i] = theY;

gX[i] = MerX[i];
gY[i] = MerY[i];

JUD[i] = Judge;

StarDraw(gX[i],gY[i],ClassDef[i]);

}
}

//クリック拡大
this.onMouseUp = function(){

_global.moveX = 0;
_global.moveY = 0;
var wakuX = 0;
var wakuY = 0;
var upX1 = 0;
var upY1 = 0;

if (_root._xmouse < Xsize && _root._ymouse < Ysize) {

if (Judge == 2) {
Judge = 3;
} else {
Judge = 1;
}

DrawJ1 = 0 ;
DrawJ2 = 0 ;
DrawJ3 = 0 ;
DrawJ4 = 0 ;
DrawJ5 = 0 ;

moveX = _root._xmouse;
moveY = _root._ymouse;

```

```

this._bmd.fillRect(new Rectangle(0,0,Xsize,Ysize),StarC2);
var _mc:MovieClip = this.createEmptyMovieClip("target_mc", this.getNextHighestDepth());
_mc.lineStyle(2 ,StarC2);
// ステージ中央を中心に半径 100 ピクセルの正円を 6 分割の近似曲線で描画

var StartY = 0;
var HB = 200 ;
for (i = 0; i < 6; i++) {
this.lineStyle(HB,COL2);
this.moveTo(0,StartY);
this.lineTo(Xsize,StartY);
StartY = StartY + HB ;

}

//星座名
WNA0.text=" ";
WNA1.text=" ";
WNA2.text=" ";
WNA3.text=" ";
WNA4.text=" ";
WNA5.text=" ";
WNA6.text=" ";
WNA7.text=" ";
WNA8.text=" ";
WNA9.text=" ";
WNA10.text=" ";
WNA11.text=" ";
WNA12.text=" ";
WNA13.text=" ";
WNA14.text=" ";
WNA15.text=" ";
WNA16.text=" ";
WNA17.text=" ";
WNA18.text=" ";
WNA19.text=" ";
WNA20.text=" ";
WNA21.text=" ";
WNA22.text=" ";
WNA23.text=" ";
WNA24.text=" ";

```

WNA25.text=" ";  
WNA26.text=" ";  
WNA27.text=" ";  
WNA28.text=" ";  
WNA29.text=" ";  
WNA30.text=" ";  
WNA31.text=" ";  
WNA32.text=" ";  
WNA33.text=" ";  
WNA34.text=" ";  
WNA35.text=" ";  
WNA36.text=" ";  
WNA37.text=" ";  
WNA38.text=" ";  
WNA39.text=" ";  
WNA40.text=" ";  
WNA41.text=" ";  
WNA42.text=" ";  
WNA43.text=" ";  
WNA44.text=" ";  
WNA45.text=" ";  
WNA46.text=" ";  
WNA47.text=" ";  
WNA48.text=" ";  
WNA49.text=" ";  
WNA50.text=" ";  
WNA51.text=" ";  
WNA52.text=" ";  
WNA53.text=" ";  
WNA54.text=" ";  
WNA55.text=" ";  
WNA56.text=" ";  
WNA57.text=" ";  
WNA58.text=" ";  
WNA59.text=" ";  
WNA60.text=" ";  
WNA61.text=" ";  
WNA62.text=" ";  
WNA63.text=" ";  
WNA64.text=" ";  
WNA65.text=" ";

```
WNA66.text=" ";
WNA67.text=" ";
WNA68.text=" ";
WNA69.text=" ";
WNA70.text=" ";
WNA71.text=" ";
WNA72.text=" ";
WNA73.text=" ";
WNA74.text=" ";
WNA75.text=" ";
WNA76.text=" ";
WNA77.text=" ";
WNA78.text=" ";
WNA79.text=" ";
WNA80.text=" ";
WNA81.text=" ";
WNA82.text=" ";
WNA83.text=" ";
WNA84.text=" ";
WNA85.text=" ";
```

//星名

```
SN0.text=" ";
SN1.text=" ";
SN2.text=" ";
SN3.text=" ";
SN4.text=" ";
SN5.text=" ";
SN6.text=" ";
SN7.text=" ";
SN8.text=" ";
SN9.text=" ";
SN10.text=" ";
SN11.text=" ";
SN12.text=" ";
SN13.text=" ";
SN14.text=" ";
SN15.text=" ";
SN16.text=" ";
```

```

SN17.text=" ";
SN18.text=" ";
SN19.text=" ";
SN20.text=" ";

xDrawCircle(_mc, Xsize/2, Ysize/2, (Ysize/2) , 6);
wakuX = moveX + ( Xsize/4 );
wakuY = moveY + ( Ysize/4 );

/*if ( moveX > ( Xsize - ( Xsize/4 ) ) ) {
if (Judge == 1) {
Judge = 4;
}
var overX = 0;
overX = wakuX - Xsize;
trace(overX);
for (var i = 0; i < gX.length; i++) {
if (0 < gX[i] && gX[i] < ( overX * 4 ) && moveY < gY[i] && gY[i] < wakuY) {
upX1 = ( gX[i] + Xsize - ( 4 * overX ) -100 ) * 4 ;
upY1 = ( gY[i] - moveY ) * 4 ;
JUD[i] = Judge;
UX[i] = upX1;
UY[i] = upY1;
gX[i] = UX[i];
gY[i] = UY[i];
StarDraw(gX[i],gY[i],ClassDef[i]);
//trace(gX[i]);
}
}
}
*/

for (var i = 0; i < gX.length; i++) {
if (moveX < gX[i] && gX[i] < wakuX && moveY < gY[i] && gY[i] < wakuY) {
upX1 = (gX[i] - moveX) * 4 ;
upY1 = (gY[i] - moveY) * 4 ;
if ( starName[i] == SNM) {
StarC = 0xFFFF0000;
} else {
StarC = 0xFFFFFFFF;
}
}
}

```



```

JUD[i] = Judge;

gX[i] = upX1;
gY[i] = upY1;

StarDraw(gX[i],gY[i],ClassDef[i]);
createTextField("TF"+i, i+1, upX1, upY1, 30, 20);
this["TF"+i].multiline = true;
this["TF"+i].textColor = 0xFF00FF00;

}
}
}

}

```

//名前数え上げ

```

function NameCount() {
_global.NaCou = [ (starName.length) ];
var k=0;
var j=0;
for (var i=0 ; i < starName.length ; i++) {

for (j=0 ; j < NaCou.length && NaCou[j] != starName[i] ; j++) ;

if(j == NaCou.length) {
NaCou[k] = starName[i];
k++;

}
}
}

```

//現代星座線描画

```

function seizaLine (a:Number,b:Number,c:Number,d:Number,e:Number,f:Number):BitmapData{
if (e == Judge && f == Judge) {
this.lineStyle(4,COL1);
this.moveTo(a,b);

```

```
this.lineTo(c,d);
return ;
```

```
}
}
```

```
//現代星座線描画 2
```

```
function seizaLine3 (a:Number,b:Number,c:Number,d:Number,e:Number,f:Number):BitmapData{
if (e == Judge && f == Judge) {
this.lineStyle(4,COL7);
this.moveTo(a,b);
this.lineTo(c,d);
return ;
```

```
}
}
```

```
//古代中国星座線描画
```

```
function seizaLine1 (a:Number,b:Number,c:Number,d:Number,e:Number,f:Number):BitmapData{
if (e == Judge && f == Judge) {
this.lineStyle(4,COL5);
this.moveTo(a,b);
this.lineTo(c,d);
return ;
```

```
}
}
```

```
//古代日本星座線描画
```

```
function seizaLine2 (a:Number,b:Number,c:Number,d:Number,e:Number,f:Number):BitmapData{
if (e == Judge && f == Judge) {
this.lineStyle(4,COL6);
this.moveTo(a,b);
this.lineTo(c,d);
return ;
```

```
}
}
```

```
//グリッド線描画
```

```
function GLine (a:Number,b:Number,c:Number,d:Number):BitmapData{
```

```

this.lineStyle(1,COL4);
this.moveTo(a,b);
this.lineTo(c,d);
return ;

}

//星描画
function StarDraw (a:Number,b:Number,c:Number):BitmapData{
this._bmd.fillRect(new Rectangle(a,b,c,c),StarC1);
return ;

}

//星消去
function ClearDraw (a:Number,b:Number,c:Number,d:Number):BitmapData{
this._bmd.fillRect(new Rectangle(a,b,c,d),StarC2);
return ;

}

//円描画
function xDrawCircle(target_mc:MovieClip, nX:Number, nY:Number, nR:Number, nSegments:Num
    if (nSegments == undefined) { // 分割数の引数省略時
        nSegments = 8; // デフォルト値設定

    }
    var nAngle:Number = 2*Math.PI/nSegments; // 分割した角度(ラジアン)
    target_mc.moveTo(nX+nR, nY);
    for (var i = 1; i<=nSegments; ++i) { // 指定数に分割して弧を描画
        var nTheta:Number = i*nAngle; // 回転角(ラジアン)
        // アンカーポイントの座標
        var nAnchorX:Number = nR*Math.cos(nTheta);
        var nAnchorY:Number = nR*Math.sin(nTheta);
        // コントロールポイントの座標
        var nControlX:Number =
            nAnchorX+nR*Math.tan(nAngle/2)*Math.cos(nTheta-Math.PI/2);
        var nControlY:Number =
            nAnchorY+nR*Math.tan(nAngle/2)*Math.sin(nTheta-Math.PI/2);
        // 弧の描画
        target_mc.curveTo(nControlX+nX, nControlY+nY, nAnchorX+nX, nAnchorY+nY);

```

```
    }  
}
```

```
//ラジアン X
```

```
function radGetX(zikanX:Number,dosuuY:Number):Number{  
var Rmax = 90;  
var R = (Rmax-dosuuY) / Rmax ;  
var h = 6 ;  
var radX = Math.cos( (zikanX * 360 / 24 * Math.PI * 2 / 360) + (h * 360 / 24 * Math.PI  
return(radX);
```

```
}
```

```
//ラジアン Y
```

```
function radGetY(zikanX:Number,dosuuY:Number):Number{  
var Rmax = 90;  
var R = (Rmax-dosuuY) / Rmax ;  
var h = 6 ;  
var radY = Math.sin( (zikanX * 360 / 24 * Math.PI * 2 / 360) + (h * 360 / 24 * Math.PI  
return(radY);
```

```
}
```

## メルカトル図法ボタンのプログラム

```
on (release) {  
  Judge = 0;  
  var StartY = 0;  
  var HB = 200 ;  
  for (i = 0; i < 6; i++) {  
    this.lineStyle(HB,COL2);  
    this.moveTo(0,StartY);  
    this.lineTo(Xsize,StartY);  
    StartY = StartY + HB ;  
  }  
}
```

//星座名

```
WNA0.text=" ";  
WNA1.text=" ";  
WNA2.text=" ";  
WNA3.text=" ";  
WNA4.text=" ";  
WNA5.text=" ";  
WNA6.text=" ";  
WNA7.text=" ";  
WNA8.text=" ";  
WNA9.text=" ";  
WNA10.text=" ";  
WNA11.text=" ";  
WNA12.text=" ";  
WNA13.text=" ";  
WNA14.text=" ";  
WNA15.text=" ";  
WNA16.text=" ";  
WNA17.text=" ";  
WNA18.text=" ";  
WNA19.text=" ";  
WNA20.text=" ";  
WNA21.text=" ";  
WNA22.text=" ";  
WNA23.text=" ";  
WNA24.text=" ";  
WNA25.text=" ";  
WNA26.text=" ";
```

WNA27.text=" ";  
WNA28.text=" ";  
WNA29.text=" ";  
WNA30.text=" ";  
WNA31.text=" ";  
WNA32.text=" ";  
WNA33.text=" ";  
WNA34.text=" ";  
WNA35.text=" ";  
WNA36.text=" ";  
WNA37.text=" ";  
WNA38.text=" ";  
WNA39.text=" ";  
WNA40.text=" ";  
WNA41.text=" ";  
WNA42.text=" ";  
WNA43.text=" ";  
WNA44.text=" ";  
WNA45.text=" ";  
WNA46.text=" ";  
WNA47.text=" ";  
WNA48.text=" ";  
WNA49.text=" ";  
WNA50.text=" ";  
WNA51.text=" ";  
WNA52.text=" ";  
WNA53.text=" ";  
WNA54.text=" ";  
WNA55.text=" ";  
WNA56.text=" ";  
WNA57.text=" ";  
WNA58.text=" ";  
WNA59.text=" ";  
WNA60.text=" ";  
WNA61.text=" ";  
WNA62.text=" ";  
WNA63.text=" ";  
WNA64.text=" ";  
WNA65.text=" ";  
WNA66.text=" ";  
WNA67.text=" ";

```
WNA68.text=" ";
WNA69.text=" ";
WNA70.text=" ";
WNA71.text=" ";
WNA72.text=" ";
WNA73.text=" ";
WNA74.text=" ";
WNA75.text=" ";
WNA76.text=" ";
WNA77.text=" ";
WNA78.text=" ";
WNA79.text=" ";
WNA80.text=" ";
WNA81.text=" ";
WNA82.text=" ";
WNA83.text=" ";
WNA84.text=" ";
WNA85.text=" ";
```

//星名

```
SN0.text=" ";
SN1.text=" ";
SN2.text=" ";
SN3.text=" ";
SN4.text=" ";
SN5.text=" ";
SN6.text=" ";
SN7.text=" ";
SN8.text=" ";
SN9.text=" ";
SN10.text=" ";
SN11.text=" ";
SN12.text=" ";
SN13.text=" ";
SN14.text=" ";
SN15.text=" ";
SN16.text=" ";
SN17.text=" ";
SN18.text=" ";
```

```

SN19.text=" ";
SN20.text=" ";

ClearDraw(0,0,Xsize,Ysize);
var _mc:MovieClip = this.createEmptyMovieClip("target_mc", this.getNextHighestDepth());
_mc.lineStyle(2 ,StarC2);
xDrawCircle(_mc, Xsize/2, Ysize/2, (Ysize/2) , 6);
for (var i = 0; i < gX.length; i++) {
JUD[i] = Judge;
gX[i] = MerX[i];
gY[i] = MerY[i];
StarDraw(gX[i],gY[i],ClassDef[i]);
}

DrawJ1 = 0 ;
DrawJ2 = 0 ;
DrawJ3 = 0 ;
DrawJ4 = 0 ;
DrawJ5 = 0 ;

}

```



## 円座標ボタンのプログラム

```
on (release) {  
  Judge = 2;  
  var StartY = 0;  
  var HB = 200 ;  
  for (i = 0; i < 6; i++) {  
    this.lineStyle(HB,COL2);  
    this.moveTo(0,StartY);  
    this.lineTo(Xsize,StartY);  
    StartY = StartY + HB ;  
  }  
}
```

//星座名

```
WNA0.text=" ";  
WNA1.text=" ";  
WNA2.text=" ";  
WNA3.text=" ";  
WNA4.text=" ";  
WNA5.text=" ";  
WNA6.text=" ";  
WNA7.text=" ";  
WNA8.text=" ";  
WNA9.text=" ";  
WNA10.text=" ";  
WNA11.text=" ";  
WNA12.text=" ";  
WNA13.text=" ";  
WNA14.text=" ";  
WNA15.text=" ";  
WNA16.text=" ";  
WNA17.text=" ";  
WNA18.text=" ";  
WNA19.text=" ";  
WNA20.text=" ";  
WNA21.text=" ";  
WNA22.text=" ";  
WNA23.text=" ";  
WNA24.text=" ";  
WNA25.text=" ";  
WNA26.text=" ";
```

WNA27.text=" ";  
WNA28.text=" ";  
WNA29.text=" ";  
WNA30.text=" ";  
WNA31.text=" ";  
WNA32.text=" ";  
WNA33.text=" ";  
WNA34.text=" ";  
WNA35.text=" ";  
WNA36.text=" ";  
WNA37.text=" ";  
WNA38.text=" ";  
WNA39.text=" ";  
WNA40.text=" ";  
WNA41.text=" ";  
WNA42.text=" ";  
WNA43.text=" ";  
WNA44.text=" ";  
WNA45.text=" ";  
WNA46.text=" ";  
WNA47.text=" ";  
WNA48.text=" ";  
WNA49.text=" ";  
WNA50.text=" ";  
WNA51.text=" ";  
WNA52.text=" ";  
WNA53.text=" ";  
WNA54.text=" ";  
WNA55.text=" ";  
WNA56.text=" ";  
WNA57.text=" ";  
WNA58.text=" ";  
WNA59.text=" ";  
WNA60.text=" ";  
WNA61.text=" ";  
WNA62.text=" ";  
WNA63.text=" ";  
WNA64.text=" ";  
WNA65.text=" ";  
WNA66.text=" ";  
WNA67.text=" ";

```
WNA68.text=" ";
WNA69.text=" ";
WNA70.text=" ";
WNA71.text=" ";
WNA72.text=" ";
WNA73.text=" ";
WNA74.text=" ";
WNA75.text=" ";
WNA76.text=" ";
WNA77.text=" ";
WNA78.text=" ";
WNA79.text=" ";
WNA80.text=" ";
WNA81.text=" ";
WNA82.text=" ";
WNA83.text=" ";
WNA84.text=" ";
WNA85.text=" ";
```

//星名

```
SN0.text=" ";
SN1.text=" ";
SN2.text=" ";
SN3.text=" ";
SN4.text=" ";
SN5.text=" ";
SN6.text=" ";
SN7.text=" ";
SN8.text=" ";
SN9.text=" ";
SN10.text=" ";
SN11.text=" ";
SN12.text=" ";
SN13.text=" ";
SN14.text=" ";
SN15.text=" ";
SN16.text=" ";
SN17.text=" ";
SN18.text=" ";
```

```

SN19.text=" ";
SN20.text=" ";

ClearDraw(0,0,Xsize,Ysize);
var _mc:MovieClip = this.createEmptyMovieClip("target_mc", this.getNextHighestDepth());
_mc.lineStyle(2, StarC1); // 線: 2ピクセル/青
// ステージ中央を中心に半径 100 ピクセルの正円を 6 分割の近似曲線で描画
xDrawCircle(_mc, Xsize/2, Ysize/2, (Ysize/2) , 6);

var hosiX = 0;
var hosiY = 0;
for (var i = 0; i < gX.length; i++) {
if (fY[i] >= 0){
JUD[i] = Judge;
hosiX = radGetX(fX[i],fY[i]) * (-1) * HsizeY + HsizeX;
hosiY = radGetY(fX[i],fY[i]) * (-1) * HsizeY + HsizeY;

gX[i] = hosiX;
gY[i] = hosiY;

StarDraw(gX[i],gY[i],ClassDef[i]);
} else {
gX[i] = 0 ;
gY[i] = 0 ;

}
}

DrawJ1 = 0 ;
DrawJ2 = 0 ;
DrawJ3 = 0 ;
DrawJ4 = 0 ;
DrawJ5 = 0 ;

}

```

## 星消去ボタンのプログラム

```
on (release) {  
  ClearDraw(0,0,Xsize,Ysize);  
  
  var _mc:MovieClip = this.createEmptyMovieClip("target_mc", this.getNextHighestDepth());  
  _mc.lineStyle(2 ,StarC2);  
  // ステージ中央を中心に半径 100 ピクセルの正円を 6 分割の近似曲線で描画  
  xDrawCircle(_mc, Xsize/2, Ysize/2, (Ysize/2) , 6);  
  
}
```

## グリッド線図示ボタンのプログラム

```
on (release) {  
  GLine(HsizeX,0,HsizeX,Ysize);  
  GLine(0,HsizeY,Xsize,HsizeY);  
  
}
```

## 現代の 88 星座ボタンのプログラム

```
//ori
seizaLine(gX[807],gY[807],gX[819],gY[819],JUD[807],JUD[819]);
seizaLine(gX[819],gY[819],gX[589],gY[589],JUD[819],JUD[589]);
seizaLine(gX[589],gY[589],gX[9],gY[9],JUD[589],JUD[9]);
seizaLine(gX[9],gY[9],gX[53],gY[53],JUD[9],JUD[53]);
seizaLine(gX[53],gY[53],gX[56],gY[56],JUD[53],JUD[56]);
seizaLine(gX[56],gY[56],gX[6],gY[6],JUD[56],JUD[6]);
seizaLine(gX[6],gY[6],gX[67],gY[67],JUD[6],JUD[67]);
seizaLine(gX[67],gY[67],gX[25],gY[25],JUD[67],JUD[25]);
seizaLine(gX[25],gY[25],gX[574],gY[574],JUD[25],JUD[574]);
seizaLine(gX[574],gY[574],gX[9],gY[9],JUD[574],JUD[9]);
seizaLine(gX[25],gY[25],gX[558],gY[558],JUD[25],JUD[558]);
seizaLine(gX[558],gY[558],gX[762],gY[762],JUD[558],JUD[762]);
seizaLine(gX[762],gY[762],gX[209],gY[209],JUD[762],JUD[209]);
seizaLine(gX[209],gY[209],gX[356],gY[356],JUD[209],JUD[356]);
seizaLine(gX[356],gY[356],gX[366],gY[366],JUD[356],JUD[366]);
seizaLine(gX[366],gY[366],gX[867],gY[867],JUD[366],JUD[867]);
.
.
.
```

以下、88 星座分のプログラムが入る。  
古代日本、中国も同様の方法である。

## 現代の行動 12 星座ボタンのプログラム (ON)

```
on (release) {  
  DrawJ1 = 1 ;  
  DrawJ2 = DrawJ2;  
  DrawJ3 = DrawJ3;  
  DrawJ4 = DrawJ4;  
  DrawJ5 = DrawJ5;  
  
  if (DrawJ1 == 0 || DrawJ2 == 0 || DrawJ3 == 0 || DrawJ4 == 0 || DrawJ5 == 0) {  
    var StartY = 0;  
    var HB = 200 ;  
    for (i = 0; i < 6; i++) {  
      this.lineStyle(HB,COL2);  
      this.moveTo(0,StartY);  
      this.lineTo(Xsize,StartY);  
      StartY = StartY + HB ;  
    }  
  }  
  .  
  .  
  .  
}
```



## 星座線消去ボタンのプログラム

```
on (release) {  
var StartY = 0;  
var HB = 200 ;  
for (i = 0; i < 6; i++) {  
this.lineStyle(HB,COL2);  
this.moveTo(0,StartY);  
this.lineTo(Xsize,StartY);  
StartY = StartY + HB ;  
  
}  
  
DrawJ1 = 0 ;  
DrawJ2 = 0 ;  
DrawJ3 = 0 ;  
DrawJ4 = 0 ;  
DrawJ5 = 0 ;  
  
}
```

## 星座名ボタンのプログラム

名前を表示させるボタンは全てこの方法でプログラムしているので、割愛させていただきます。

```
on (release) {
for (var i = 0 ; i < NaCou.length ; i++) {
var Xsum = 0;
var Ysum = 0;
var Xave=0;
var Yave=0;
var k=0;
for(var j =0 ; j < starName.length ; j++){
if (NaCou[i] == starName[j]){
if (j == 508 ||
j == 940 ||
j == 918 ||
j == 952 ||
j == 927 ||
j == 417 ||
j == 604 ||
j == 712 ||
j == 336 ||
j == 816 ||
j == 527 ||
j == 598 ||
j == 698 ||
j == 937 ||
j == 358 ||
j == 902 ||
j == 137 ||
j == 718
) {
Xsum = Xsum+0;
Ysum = Ysum+0;
} else {
if (Judge == JUD[j]) {
Xsum = Xsum+gX[j];
Ysum = Ysum+gY[j];
k++;
}
}
}
```

```

}
}
if (k != 0) {
Xave = Xsum/k;
Yave = Ysum/k;
} else {
Xave=0;
Yave=0;

}
createTextField("WNA"+i, i+1, Xave, Yave ,120 ,20);
this["WNA"+i].multiline = true;
if (Xave == 0 || Yave == 0 ) {
this["WNA"+i].textColor = 0xFF000000;
} else {
this["WNA"+i].textColor = 0xFF00FF00;

}
}
}
ONA0.text=" ";
ONA1.text=" ";
ONA2.text=" ";
ONA3.text=" ";
ONA4.text=" ";
ONA5.text=" ";
ONA6.text=" ";
ONA7.text=" ";
ONA8.text=" ";
ONA9.text=" ";
ONA10.text=" ";
ONA11.text=" ";
ONA12.text=" ";
ONA13.text=" ";
ONA14.text=" ";
ONA15.text=" ";
ONA16.text=" ";
ONA17.text=" ";
ONA18.text=" ";
ONA19.text=" ";
ONA20.text=" ";
ONA21.text=" ";

```

```
ONA22.text=" ";
ONA23.text=" ";
ONA24.text=" ";
ONA25.text=" ";
```

```
SN0.text=" ";
SN1.text=" ";
SN2.text=" ";
SN3.text=" ";
SN4.text=" ";
SN5.text=" ";
SN6.text=" ";
SN7.text=" ";
SN8.text=" ";
SN9.text=" ";
SN10.text=" ";
SN11.text=" ";
SN12.text=" ";
SN13.text=" ";
SN14.text=" ";
SN15.text=" ";
SN16.text=" ";
SN17.text=" ";
SN18.text=" ";
SN19.text=" ";
SN20.text=" ";
```

```
WNA0.text=NaCou[0];
WNA1.text=NaCou[1];
WNA2.text=NaCou[2];
WNA3.text=NaCou[3];
WNA4.text=NaCou[4];
WNA5.text=NaCou[5];
WNA6.text=NaCou[6];
WNA7.text=NaCou[7];
WNA8.text=NaCou[8];
WNA9.text=NaCou[9];
WNA10.text=NaCou[10];
WNA11.text=NaCou[11];
WNA12.text=NaCou[12];
WNA13.text=NaCou[13];
```

WNA14.text=NaCou[14];  
WNA15.text=NaCou[15];  
WNA16.text=NaCou[16];  
WNA17.text=NaCou[17];  
WNA18.text=NaCou[18];  
WNA19.text=NaCou[19];  
WNA20.text=NaCou[20];  
WNA21.text=NaCou[21];  
WNA22.text=NaCou[22];  
WNA23.text=NaCou[23];  
WNA24.text=NaCou[24];  
WNA25.text=NaCou[25];  
WNA26.text=NaCou[26];  
WNA27.text=NaCou[27];  
WNA28.text=NaCou[28];  
WNA29.text=NaCou[29];  
WNA30.text=NaCou[30];  
WNA31.text=NaCou[31];  
WNA32.text=NaCou[32];  
WNA33.text=NaCou[33];  
WNA34.text=NaCou[34];  
WNA35.text=NaCou[35];  
WNA36.text=NaCou[36];  
WNA37.text=NaCou[37];  
WNA38.text=NaCou[38];  
WNA39.text=NaCou[39];  
WNA40.text=NaCou[40];  
WNA41.text=NaCou[41];  
WNA42.text=NaCou[42];  
WNA43.text=NaCou[43];  
WNA44.text=NaCou[44];  
WNA45.text=NaCou[45];  
WNA46.text=NaCou[46];  
WNA47.text=NaCou[47];  
WNA48.text=NaCou[48];  
WNA49.text=NaCou[49];  
WNA50.text=NaCou[50];  
WNA51.text=NaCou[51];  
WNA52.text=NaCou[52];  
WNA53.text=NaCou[53];  
WNA54.text=NaCou[54];

WNA55.text=NaCou[55];  
WNA56.text=NaCou[56];  
WNA57.text=NaCou[57];  
WNA58.text=NaCou[58];  
WNA59.text=NaCou[59];  
WNA60.text=NaCou[60];  
WNA61.text=NaCou[61];  
WNA62.text=NaCou[62];  
WNA63.text=NaCou[63];  
WNA64.text=NaCou[64];  
WNA65.text=NaCou[65];  
WNA66.text=NaCou[66];  
WNA67.text=NaCou[67];  
WNA68.text=NaCou[68];  
WNA69.text=NaCou[69];  
WNA70.text=NaCou[70];  
WNA71.text=NaCou[71];  
WNA72.text=NaCou[72];  
WNA73.text=NaCou[73];  
WNA74.text=NaCou[74];  
WNA75.text=NaCou[75];  
WNA76.text=NaCou[76];  
WNA77.text=NaCou[77];  
WNA78.text=NaCou[78];  
WNA79.text=NaCou[79];  
WNA80.text=NaCou[80];  
WNA81.text=NaCou[81];  
WNA82.text=NaCou[82];  
WNA83.text=NaCou[83];  
WNA84.text=NaCou[84];  
WNA85.text=NaCou[85];

}

## 全消去ボタンのプログラム

```
on (release) {
var StartY = 0;
var HB = 200 ;
for (i = 0; i < 6; i++) {
this.lineStyle(HB,COL2);
this.moveTo(0,StartY);
this.lineTo(Xsize,StartY);
StartY = StartY + HB ;

}
ClearDraw(0,0,Xsize,Ysize);

var _mc:MovieClip = this.createEmptyMovieClip("target_mc", this.getNextHighestDepth());
_mc.lineStyle(2 ,StarC2);
// ステージ中央を中心に半径 100 ピクセルの正円を 6 分割の近似曲線で描画
xDrawCircle(_mc, Xsize/2, Ysize/2, (Ysize/2) , 6);

//星座名
WNA0.text=" ";
WNA1.text=" ";
WNA2.text=" ";
WNA3.text=" ";
WNA4.text=" ";
WNA5.text=" ";
WNA6.text=" ";
WNA7.text=" ";
WNA8.text=" ";
WNA9.text=" ";
WNA10.text=" ";
WNA11.text=" ";
WNA12.text=" ";
WNA13.text=" ";
WNA14.text=" ";
WNA15.text=" ";
WNA16.text=" ";
WNA17.text=" ";
WNA18.text=" ";
WNA19.text=" ";
WNA20.text=" ";
WNA21.text=" ";
```

WNA22.text=" ";  
WNA23.text=" ";  
WNA24.text=" ";  
WNA25.text=" ";  
WNA26.text=" ";  
WNA27.text=" ";  
WNA28.text=" ";  
WNA29.text=" ";  
WNA30.text=" ";  
WNA31.text=" ";  
WNA32.text=" ";  
WNA33.text=" ";  
WNA34.text=" ";  
WNA35.text=" ";  
WNA36.text=" ";  
WNA37.text=" ";  
WNA38.text=" ";  
WNA39.text=" ";  
WNA40.text=" ";  
WNA41.text=" ";  
WNA42.text=" ";  
WNA43.text=" ";  
WNA44.text=" ";  
WNA45.text=" ";  
WNA46.text=" ";  
WNA47.text=" ";  
WNA48.text=" ";  
WNA49.text=" ";  
WNA50.text=" ";  
WNA51.text=" ";  
WNA52.text=" ";  
WNA53.text=" ";  
WNA54.text=" ";  
WNA55.text=" ";  
WNA56.text=" ";  
WNA57.text=" ";  
WNA58.text=" ";  
WNA59.text=" ";  
WNA60.text=" ";  
WNA61.text=" ";  
WNA62.text=" ";



```
WNA63.text=" ";
WNA64.text=" ";
WNA65.text=" ";
WNA66.text=" ";
WNA67.text=" ";
WNA68.text=" ";
WNA69.text=" ";
WNA70.text=" ";
WNA71.text=" ";
WNA72.text=" ";
WNA73.text=" ";
WNA74.text=" ";
WNA75.text=" ";
WNA76.text=" ";
WNA77.text=" ";
WNA78.text=" ";
WNA79.text=" ";
WNA80.text=" ";
WNA81.text=" ";
WNA82.text=" ";
WNA83.text=" ";
WNA84.text=" ";
WNA85.text=" ";
```

//星名

```
SN0.text=" ";
SN1.text=" ";
SN2.text=" ";
SN3.text=" ";
SN4.text=" ";
SN5.text=" ";
SN6.text=" ";
SN7.text=" ";
SN8.text=" ";
SN9.text=" ";
SN10.text=" ";
SN11.text=" ";
SN12.text=" ";
SN13.text=" ";
```

```
SN14.text=" ";  
SN15.text=" ";  
SN16.text=" ";  
SN17.text=" ";  
SN18.text=" ";  
SN19.text=" ";  
SN20.text=" ";  
}
```

## 特定の星座拡大ボタンのプログラム

オリオン座を特定拡大するプログラムのソースのみ添付する。

```
on (release) {  
var StartY = 0;  
var HB = 200 ;  
for (i = 0; i < 6; i++) {  
this.lineStyle(HB,COL2);  
this.moveTo(0,StartY);  
this.lineTo(Xsize,StartY);  
StartY = StartY + HB ;  
  
}
```

//星座名

```
WNA0.text=" ";  
WNA1.text=" ";  
WNA2.text=" ";  
WNA3.text=" ";  
WNA4.text=" ";  
WNA5.text=" ";  
WNA6.text=" ";  
WNA7.text=" ";  
WNA8.text=" ";  
WNA9.text=" ";  
WNA10.text=" ";  
WNA11.text=" ";  
WNA12.text=" ";  
WNA13.text=" ";  
WNA14.text=" ";  
WNA15.text=" ";  
WNA16.text=" ";  
WNA17.text=" ";  
WNA18.text=" ";  
WNA19.text=" ";  
WNA20.text=" ";  
WNA21.text=" ";  
WNA22.text=" ";  
WNA23.text=" ";  
WNA24.text=" ";  
WNA25.text=" ";  
WNA26.text=" ";
```

WNA27.text=" ";  
WNA28.text=" ";  
WNA29.text=" ";  
WNA30.text=" ";  
WNA31.text=" ";  
WNA32.text=" ";  
WNA33.text=" ";  
WNA34.text=" ";  
WNA35.text=" ";  
WNA36.text=" ";  
WNA37.text=" ";  
WNA38.text=" ";  
WNA39.text=" ";  
WNA40.text=" ";  
WNA41.text=" ";  
WNA42.text=" ";  
WNA43.text=" ";  
WNA44.text=" ";  
WNA45.text=" ";  
WNA46.text=" ";  
WNA47.text=" ";  
WNA48.text=" ";  
WNA49.text=" ";  
WNA50.text=" ";  
WNA51.text=" ";  
WNA52.text=" ";  
WNA53.text=" ";  
WNA54.text=" ";  
WNA55.text=" ";  
WNA56.text=" ";  
WNA57.text=" ";  
WNA58.text=" ";  
WNA59.text=" ";  
WNA60.text=" ";  
WNA61.text=" ";  
WNA62.text=" ";  
WNA63.text=" ";  
WNA64.text=" ";  
WNA65.text=" ";  
WNA66.text=" ";  
WNA67.text=" ";

```
WNA68.text=" ";
WNA69.text=" ";
WNA70.text=" ";
WNA71.text=" ";
WNA72.text=" ";
WNA73.text=" ";
WNA74.text=" ";
WNA75.text=" ";
WNA76.text=" ";
WNA77.text=" ";
WNA78.text=" ";
WNA79.text=" ";
WNA80.text=" ";
WNA81.text=" ";
WNA82.text=" ";
WNA83.text=" ";
WNA84.text=" ";
WNA85.text=" ";
```

//星名

```
SN0.text=" ";
SN1.text=" ";
SN2.text=" ";
SN3.text=" ";
SN4.text=" ";
SN5.text=" ";
SN6.text=" ";
SN7.text=" ";
SN8.text=" ";
SN9.text=" ";
SN10.text=" ";
SN11.text=" ";
SN12.text=" ";
SN13.text=" ";
SN14.text=" ";
SN15.text=" ";
SN16.text=" ";
SN17.text=" ";
SN18.text=" ";
```

```

SN19.text=" ";
SN20.text=" ";

ClearDraw(0,0,Xsize,Ysize);

var _mc:MovieClip = this.createEmptyMovieClip("target_mc", this.getNextHighestDepth());
_mc.lineStyle(2 ,StarC2);
// ステージ中央を中心に半径 100 ピクセルの正円を 6 分割の近似曲線で描画
xDrawCircle(_mc, Xsize/2, Ysize/2, (Ysize/2) , 6);

minX = MerX[6];
minY = MerY[6];
Judge = 3;
for (var i = 0 ; i < starName.length ; i++ ) {
if ( starName[i] == "Ori" ) {
if (minX > MerX[i]) {
minX = MerX[i];
}
if (minY > MerY[i]) {
minY = MerY[i];
}
}
}

for (var i = 0 ; i < starName.length ; i++ ) {
if ( starName[i] == "Ori" ) {
JUD[i] = Judge;
gX[i] = ((MerX[i] - minX) * 4 ) + HsizeX;
gY[i] = (MerY[i] - minY) * 4;
StarDraw(gX[i] ,gY[i] ,ClassDef[i]);
} else {
gX[i] = 0;
gY[i] = 0;
}
}
}

```