

Fly! II Gauge Reference Guide

Version: Draft 0.1

Editor: Chris Wallace c_wallace@sympatico.ca

Contributors: Richard Harvey
Paul Russell
Jennifer Goss
Chris Wallace

Introduction

This reference guide contains detailed descriptions of all gauge objects in Fly! II. Much of the information presented here was derived either from the IDK files written by Paul Russell and Jennifer Goss, or from unpublished gauge reference notes for Fly! written by Richard Harvey. For those objects which had no documentation available from TRI, the internals have been deduced by looking at examples of the gauge types in TRI's default aircraft.

This reference guide assumes that the reader is familiar with Fly! stream files and the general architecture of Fly! subsystems and the various graphic file formats used by Fly!

TODO: Separate document with overview of panel system, interactive panels vs. static views, panel lighting, etc.

Gauge Operational Overview

In the context of this document, a "Gauge Object" represents all objects which are specified in the gauge portion of a Fly! aircraft panel file. These files are located in the \Data subfolder of the aircraft POD, and have extensions .PNL, .VFR, .VFN, .IFR or .IFN. Gauge objects refer to anything on a panel that either displays information and/or receives user input via mouse clicks.

There are two different types of behaviour associated with gauge objects, depending on whether user interaction is a part of the gauge's behaviour or not. Gauges which only display information, for example airspeed needle gauges, EFIS primary flight display, etc. are examples of non-interactive gauges. Gauges which react to user mouse clicks on various parts of the gauge such as pushbutton or toggle switches, are examples of interactive gauges.

The behaviour of custom DLL gauges provides an indication of how the internal gauges operate. The first time a particular panel is displayed, the specification file (.PNL, .VFR, .VFN, .IFR or .IFN) is read and for each gauge object in the file, an instance of the appropriate class is instantiated. All of the "Read Tags" within the <bgno> <endo> pair in the panel file are passed to the gauge instance, so that it can update its internal state variables depending on how the aircraft designer wants the gauge to appear and behave.

Every time the screen is refreshed (typically around 10-30 times per second), Fly! determines which gauge objects are actually visible on the user's screen. This will be influenced by: (a) whether the current view is an internal (cockpit) view or an external aircraft view, (b) if an internal view, which panel (.PNL etc.) or static view (.480 etc.) is active, and (c) if a panel, which gauges on the panel are visible based on the scrolling position, screen size, etc.

For every gauge that is visible (even partially visible), the Draw member function of the gauge class (or in the case of a custom DLL gauge, the DLLDraw callback) is called by Fly!. Each gauge object is responsible for updating its internal state and re-drawing its “appearance” on the panel. So if a gauge is not actually visible on the screen, then its Draw function will not be called. It should be pointed out that this illustrates a fundamental point about the Fly! architecture : that a gauge object’s only role in life is to provide a visual representation of a data value; it is not responsible for actually calculating any simulation state values. That is the job of the Subsystem Objects specified in the various subsystem specification files (.AMP, .ENG, etc.)

The Draw function for non-interactive gauges would typically be fairly straightforward. The gauge first determines the data values of all subsystems that it is dependent upon. For example, a dual-needle gauge which is displaying the engine RPM for a twin-engine aircraft might need to retrieve the actual RPM values from two ‘erpm’ subsystem objects. These details are hard-coded into the Draw function of the CTwoNeedleGauge class; rather the subsystems would be specified within the <ndl1> and <ndl2> data tags in the ‘2ned’ gauge object. The Draw function in the gauge instance would send a GETDATA message to each of the two subsystems, and use the values returned by the subsystems as the raw data to be displayed.

After the gauge has determined the raw data value, it then uses this raw data to actually render the gauge onto the panel surface (which was passed in as an argument to the Draw function). For example, a needle gauge would perform some mapping of the raw data value to a needle angle between 0-360 degrees, then use this angle as an index into a 360-frame bitmap image of the actual needle graphics. Another type of gauge might take the raw data value and display it digitally with an LED-like font.

Gauge Type Summary Table

Table 1 lists all gauge types in their hierarchical relationships. The most important piece of information from a designer’s point of view is the ID column. This four-character unique identifier for the gauge type is used in the panel (PNL) file to create an instance of that gauge on the aircraft panel. The Class Class refers to the internal C++ class implementation with Fly! II, you do not have to use this value anywhere.

Following Table 1 is a breakdown of all the gauges in alphabetical order by ID. Each table entry links to the detailed information page by clicking on either the signature or description for the gauge.

Type	Class	Description
gage	CGauge	Generic Gauge
bgge	CBitmapGauge	Generic Bitmap Gauge
need	CNeedleGauge	Generic Needle Gauge
dned	CDigitalNeedleGauge	Digital Needle Gauge
ghst	CGhostGauge	Ghost Gauge
aspd	CAirspeedGauge	Airspeed Indicator Needle Gauge
vspd	CVerticalSpeedGauge	Vertical Speed Indicator Needle Gauge
gyro	CDirectionalGyroGauge	Directional Gyro Needle Gauge
fuel	CFuelGauge	Fuel Quantity Needle Gauge
oilt	COilTemperatureGauge	Oil Temperature Needle Gauge
oilp	COilPressureGauge	Oil Pressure Needle Gauge
amms	CAmmeterGauge	Ammeter Needle Gauge
tach	CTachometerGauge	Tachometer Needle Gauge
vacu	CVacuumGauge	Vacuum Pressure Needle Gauge
adfg	CBasicADFGauge	Automatic Direction Finder Needle Gauge
egtg	CExhaustGasTempGauge	Exhaust Gas Temperature Needle Gauge
fflw	CFuelFlowGauge	Fuel Flow Needle Gauge
suct	CSuctionGauge	Suction Pressure Needle Gauge

namp	CJoAmpsGauge	Piper Navajo Ammeter Needle Gauge
nrar	CJoRadarAltimeterGauge	Piper Navajo Radar Altimeter Gauge
2ned	CTwoNeedleGauge	Two-Needle Gauge
dtch	CDualTachometerGauge	Dual Tachometer Gauge
aclk	CAnalogClockGauge	Analog Clock Gauge
brpg	CBrakePressureGauge	Brake Pressure Gauge
nlnq	CN1NeedleGauge	Dual N1 Tachometer Gauge
aoan	CAOANeedleGauge	Angle of Attack Needle Gauge
3ned	CThreeNeedleGauge	Three-Needle Gauge
atti	CAttitudeGauge	Attitude Indicator Gauge
hori	CHorizonGauge	Artificial Horizon Gauge
fdhz	CFDHorizonGauge	Artificial Horizon Gauge w/Flight Director
alti	CAltimeterGauge	Altimeter Gauge
altr	CRollingAltimeterGauge	Rolling Altimeter Gauge
turn	CTurnCoordinatorGauge	Turn Coordinator
slip	CSlipIndicator	Slip Indicator
hobb	CHobbsMeterGauge	Hobbs Meter
comp	CBasicCompassGauge	Whiskey Compass
navg	CNavigationGauge	Navigation (CDI) Gauge
nrdg	CNavRadioGauge	Nav Radio
kn53	CBasicKingNavRadioGauge	King KN-53 Nav Radio
kn80	CComplexKingNavRadioGauge	King KN-80 Nav Radio
comm	CCommRadioGauge	Comm Radio
ky96	CBasicKingCommRadioGauge	King KY-96 Comm Radio
kt79	CBasicKingTransponderGauge	King KT-79 Transponder
atop	CBasicKingAutopilotGauge	King Autopilot
cdcg	CFlyhawkDigitalClockGauge	Flyhawk Digital Clock
bdcg	CBasicDigitalClockGauge	Basic Digital Clock
k155	CBKNavCommKX155Gauge	Bendix-King KX-155 Nav/Comm Radio
kl89	CBKGPSKLN89Gauge	Bendix-King KLN-89 GPS
kt76	CBKXPDRKT76Gauge	Bendix-King KT-76 Transponder
kr87	CBKADFKR87Gauge	Bendix-King KR-87 ADF
kal4	CBKAP140Gauge	Bendix-King KAP-140 Autopilot
kf20	CBKFC200Gauge	Bendix-King KFC-200 Autopilot
kf15	CBKKFC150Gauge	Bendix-King KFC-150 Autopilot
kma2	CBKAudioKMA26Gauge	Bendix-King KMA-26 Audio Panel
shan	CFlyhawkAnnunciatorPanel	Flyhawk Annunciator Panel
shs	CFlyhawkAnnunciatorTestSwitch	Flyhawk Annunciator Test Switch
hsi_	CHorizontalSituationGauge	Horizontal Situation Gauge
selt	CFlyhawkELTSwitch	Flyhawk Emergency Location Transmitter
navr	CGenericNavRadioGauge	Generic Nav Radio
comr	CGenericCommRadioGauge	Generic Comm Radio
xpdr	CGenericTransponderGauge	Generic Transponder Radio
adf_	CGenericADFRadioGauge	Generic ADF Radio
rmig	CRMINeedleGauge	Remote Magnetic Indicator Gauge
dgrg	CDigitalReadoutGauge	Digital Readout Gauge
hspt	CMomentaryHotSpotGauge	Momentary Hotspot
cprk	CCabinPressureRateKnobGauge	Cabin Pressure Rate Knob
cprs	CCabinPressureGauge	Cabin Pressure Gauge
mcht	CMalybooCHTIndicatorGauge	Piper Malibu CHT Indicator
mchb	CMalybooCHTButtonGauge	Piper Malibu CHT Select Button
mngp	CMalybooNavGpsPanelGauge	Piper Malibu Nav/GPS Panel
mdcg	CMalybooDigitalClockGauge	Piper Malibu Digital Clock
gslv	CGyroSlavingGauge	Gyro Slaving Gauge
vert	CVerticalRateAltGauge	Vertical Rate Indicator
cpfd	CCollinsPFDGauge	Collins Primary Flight Display
cnvd	CCollinsNDGauge	Collins Navigation Display
crtu	CCollinsRTUGauge	Collins Radio Tuning Unit
cadd	CCollinsADGauge	Collins Air Data Display
cmdnd	CCollinsMNDGauge	Collins Multi-Nav Display
sync	CPropSyncFanGauge	Prop Synchrophaser
hbal	CHorizonBallGauge	Horizon Indicator
ufms	CUniversalFMSGauge	Universal Flight Management System
cesp	CCollinsEFISSelectionPanel	Collins EFIS Selection Panel
cvsp	CCollinsVSpeedPanel	Collins Vertical Speed Panel
khfr	CCollinsHFRadioGauge	Collins HF Radio
sngp	CFlyhawkNavGpsPanelGauge	Flyhawk Nav/GPS Panel
shfs	CFlyhawkFuelSelector	Flyhawk Fuel Selector
njfs	CJoFuelSelector	Piper Navajo Fuel Selector

shet	CFlyhawkElevatorTrim	Flyhawk Elevator Trim
mfss	CMalybooFuelSelectSlider	Piper Malibu Fuel Select Slider
pmem	CMalybooEngineMonitorGauge	Piper Malibu Engine Monitor
annc	CAnnunciatorLightGauge	Annunciator Light
dmei	CDMEIndicatorGauge	DME Indicator
dme2	CDME2IndicatorGauge	DME Indicator
wrad	CWeatherRadarGauge	Weather Radar
drog	CDigitsReadoutGauge	Digital Readout
hsat	CWalkerTemperatureAirSpeedGauge	Walker Temp/Airspeed Indicator
apre	CAnalogAltPreselectGauge	Analog Altitude Preselect
tcag	CTCASGauge	TCAS
sios	CSimpleInOutStateSwitch	Simple In/Out Switch
lios	CLitInOutStateSwitch	Lit In/Out Switch
swit	CSimpleSwitchGauge	Simple Switch
batt	CBasicBatterySwitch	Battery Switch
elt_	CTwoStateELTSwitch	Emergency Location Transmitter Switch
pair	CSwitchPairGauge	Switch Pair
fswi	CSimpleFloatSwitchGauge	Floating-Point Switch
dswi	CDualSwitchGauge	Dual Switch
knob	CPushPullKnobGauge	Push/Pull Knob
prim	CPrimerKnobGauge	Fuel Primer
flap	CFlapGauge	Flaps Lever
llgr	CLitLandingGearKnobGauge	Lit Landing Gear Knob
mixt	CMixtureKnobGauge	Mixture Knob
revt	CReversibleThrottleKnob	Reversible Throttle Knob
dual	CDualKnob	Dual Knob
shlk	CFlyhawkLightKnob	Flyhawk Light (Dual) Knob
tknb	CTurnKnobGauge	Turn Knob
magn	CBasicMagnetosSwitch	Magnetos Switch
cswi	CCoveredSwitchGauge	Covered Switch
cbtn	CCoveredLightButtonGauge	Covered Lighted Button
vsld	CVerticalSliderGauge	Vertical Slider
twhl	CTrimWheelGauge	Trim Wheel Control
twhi	CTrimWheelIndicatorGauge	Trim Wheel Indicator
indi	CIndicatorGauge	Indicator
find	CFlapsIndicatorGauge	Flaps Indicator
push	CPushButtonGauge	Push Button
lbut	CLightButtonGauge	Lighted Button
tswi	CTurnSwitchGauge	Turn Switch
mcwb	CMasterCautionButton	Master Caution/Warning Button
kafe	CKAFireExtinguisherButton	King Air Fire Extinguisher
happ	CWalkerAPPPanel	Walker Autopilot Panel
rhfp	CWalkerFirePanel	Walker Fire Panel
lssg	CLightSwitchSetGauge	Light Switch Set
rock	CRockerSwitchGauge	Rocker Switch
pefs	CPilatusEFISPanel	Pilatus EFIS Panel
ngrp	CNavRadioGroup	Nav Radio Group
cgrp	CCommRadioGroup	Comm Radio Group
ogrp	CNavGaugeGroup	Nav Gauge Group
tgrp	CTransponderRadioGroup	Transponder Radio Group
agrp	CADFRadioGroup	ADF Radio Group
radg	CRadioGaugeGroup	Radio Gauge Group

Within many of the gauge types, sub-objects are embedded for such common things as needle specifications, internal message targets, etc. These sub-objects are enclosed by <bgn0> and <end0> tags, and can contain all valid data tags for the sub-object. For example, the first gauge type in the reference section, '2ned', has a data tag of <nd12> which expects a data type of OBJECT, Needle. The corresponding object type is 'need' (Generic Needle Gauge) so an instance of this tag in a '2ned' gauge would look like:

```

<nd12>
<bgn0>
... data tags for gauge type 'need'
<end0>

```

2ned Two-Needle Gauge

Class: CTwoNeedleGauge

Base Class: CBitmapGauge

Description:

The '2ned' gauge is a simple combination of two independent indicator needles on a single gauge surface. Details of needle gauge operation are described under the entry for gauge type 'need' (Needle Gauge). The only object-specific tags are <ndl1> and <ndl2> which specify the Needle objects representing each of the two needles.

Read Tags

Signature	Description	Type	Format
ndl1	Needle 1	OBJECT	Needle
ndl2	Needle 2	OBJECT	Needle

Example (TRI200.PNL)

```
<gage> ==== GAUGE ENTRY ====
2ned
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  fflw
  <size> ---- placement and size ----
  837
  65
  95
  95
  <ndl1> ---- engine 1 rpm needle ----
  <bgno> ---- BEGIN NEEDLE ----
    <mesg> -- connect to Left Fuelflow Subsystem --
    <bgno>
      <conn>
      Lffg
      <dtag>
      indn
    <endo>
    <cntr> ---- needle center x,y ----
    47
    47
    <ned1> ---- bitmap needle ----
    stnd
    <bgno> ==== BEGIN BITMAP NEEDLE ====
      <shar> ---- shared needle ----
      <offx> ---- offset X ----
      35
      <offy> ---- offset Y ----
      35
      <base> ---- base bitmap name ----
      NEEDLE1
    <endo> ==== END BITMAP NEEDLE ====
    <gtbl> ---- gauge value table ----
    <bgno> ---- begin table ----
      <fmt1> - x y
      10.0 0.0
      18.0 38.5
      24.0 71.0
      30.0 97.0
      34.0 116.5
      38.0 137.0
```



```

45.0 164.0
<endf> - end format
<endo> ---- end table ----
<sang> ---- starting angle ----
188.0
<gmin> ---- min value allowed ----
10.0
<gmax> ---- max value allowed ----
45.0
<dfmt> ---- display format ----
%.1f gph (Left)
<endo> ---- END NEEDLE ----
<ndl2> ---- engine 2 rpm needle ----
<bgno> ---- BEGIN NEEDLE ----
<mesg> -- connect to Left Fuelflow Subsystem --
<bgno>
    <conn>
    Rffg
    <dtag>
    indn
<endo>
<cntr> ---- needle center x,y ----
47
47
<ned1> ---- bitmap needle ----
stnd
<bgno> ===== BEGIN BITMAP NEEDLE =====
    <shar> ---- shared needle ----
    <offx> ---- offset X ----
    35
    <offy> ---- offset Y ----
    35
    <base> ---- base bitmap name ----
    NEEDLE1
<endo> ===== END BITMAP NEEDLE =====
<gtbl> ---- gauge value table ----
<bgno> ---- begin table ----
    <fmt1> - x y
    10.0 164.0
    18.0 125.5
    24.0 93.0
    30.0 67.5
    34.0 46.5
    38.0 26.0
    45.0 0.0
    <endf> - end format
<endo> ---- end table ----
<sang> ---- starting angle ----
8.0
<gmin> ---- min value allowed ----
10.0
<gmax> ---- max value allowed ----
45.0
<dfmt> ---- display format ----
%.1f gph (Right)
<endo> ---- END NEEDLE ----
<lite> ---- light ----
plit
<mask> ---- mask ----
pnmask18.raw
<help> ---- popup help ----
Fuel Flow
<endo> ===== END GAUGE ENTRY =====

```

3ned Three-Needle Gauge

Class: CThreeNeedleGauge

Base Class: CBitmapGauge

Description:

The '3ned' gauge is a simple combination of three independent indicator needles on a single gauge surface. Details of needle gauge operation are described under the entry for gauge type 'need' (Needle Gauge). The only object-specific tags are <ndl1>, <ndl2> and <ndl3> which specify the Needle objects representing each of the three needles.

Read Tags

Signature	Description	Type	Format
ndl1	Needle 1	OBJECT	Needle
ndl2	Needle 2	OBJECT	Needle
ndl3	Needle 3	OBJECT	Needle

Example (TRI200.PNL)

```
<gage> ==== GAUGE ENTRY ====
3ned
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  eng1
  <size> ---- placement and size ----
  1021
  316
  90
  90
  <ndl1> ---- engine 1 oil pressure needle ----
  <bgno> ---- BEGIN NEEDLE ----
    <conn> ---- info tag ----
    oilP
    <user> ---- user data ----
    HARDWARE,GAUGE
    <user> ---- user data ----
    ENGINE,1
    <cntr> ---- needle center x,y ----
    45
    30
    <ned1> ---- needle entry ----
    tiny
    <bgno> ==== BEGIN BITMAP NEEDLE ====
      <shar> ---- shared needle ----
      <base> ---- base bitmap name ----
      needle4
    <endo> ==== END BITMAP NEEDLE ====
    <gtbl> ---- gauge value table ----
    <bgno> ---- begin table ----
      <fmt1> - x y
      0.0 0.0
      50.0 46.0
      100.0 90.0
      150.0 134.0
      200.0 180.0
      <endf> - end format
    <endo> ---- end table ----
    <sang> ---- starting angle ----
    270.0
    <gmin> ---- min value allowed ----
```



```

0.0
<gmax> ---- max value allowed ----
20.0
<dfmt> ---- display format ----
Pressure %.1f psi
<endo> ---- END NEEDLE ----
<ndl2> ---- engine 1 cyliner temp ----
<bgn> ---- BEGIN NEEDLE ----
<conn> ---- info tag ----
eCHT
<user> ---- user data ----
HARDWARE,GAUGE
<user> ---- user data ----
ENGINE,1
<cntr> ---- needle center x,y ----
23
56
<nedl> ---- needle entry ----
tiny
<bgn> ===== BEGIN BITMAP NEEDLE =====
<shar> ---- shared needle ----
<base> ---- base bitmap name ----
needle4
<endo> ===== END BITMAP NEEDLE =====
<gtbl> ---- gauge value table ----
<bgn> ---- begin table ----
<fmt1> - x y
100.0 145.0
200.0 108.0
300.0 72.0
400.0 38.0
500.0 0.0
<endf> - end format
<endo> ---- end table ----
<sang> ---- starting angle ----
18.0
<gmin> ---- min value allowed ----
100.0
<gmax> ---- max value allowed ----
500.0
<dfmt> ---- display format ----
Cylinder %.1f
<unit> ---- display units ----
FAHRENHEIT
<endo> ---- END NEEDLE ----
<ndl3> ---- engine 1 oil temp ----
<bgn> ---- BEGIN NEEDLE ----
<conn> ---- info tag ----
oilT
<user> ---- user data ----
HARDWARE,GAUGE
<user> ---- user data ----
ENGINE,1
<cntr> ---- needle center x,y ----
67
56
<nedl> ---- needle entry ----
tiny
<bgn> ===== BEGIN BITMAP NEEDLE =====
<shar> ---- shared needle ----
<base> ---- base bitmap name ----
needle4
<endo> ===== END BITMAP NEEDLE =====
<gtbl> ---- gauge value table ----
<bgn> ---- begin table ----
<fmt1> - x y
50.0 0.0
100.0 35.0
150.0 71.0
200.0 105.0
250.0 141.0

```

```

        <endf> - end format
<endo> ---- end table ----
<sang> ---- starting angle ----
199.0
<gmin> ---- min value allowed ----
50.0
<gmax> ---- max value allowed ----
250.0
<dfmt> ---- display format ----
Oil %.1f
<unit> ---- display units ----
FAHRENHEIT
<endo> ---- END NEEDLE ----
<help> ---- popup help ----
Engine #1 Oil & Temperature
<lite> ---- light ----
clit
<mask> ---- mask ----
pnmask23.raw
<endo> ==== END GAUGE ENTRY ====
```

aclk Analog Clock

Class: CAnalogClockGauge

Base Class: CTwoNeedleGauge

Description:

This gauge type represents an analog clock, with hour and minute hands (but no second hand). All data tags are inherited from its parent '2ned' Two-Needle Gauge; the <ndl1> tag specifies the needle to be used for the hour hand, and the <ndl2> tag specifies the needle to be used for the minute hand. The needle positions are automatically synchronized to the simulation time-of-day; there is no need to include a <mesg> connection to a data provider.

Read Tags

Signature	Description	Type	Format

Example (TRI200.PNL)

```
<gage> ==== Analog Clock GAUGE ENTRY ====
aclk
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  clk1
  <size> ---- x,y,xsize,ysize ----
  49
  144
  55
  55
  <ndl1> ---- bitmap needle ----
  <bgn> ==== BEGIN NEEDLE ====
    <cntr> ---- needle center x,y ----
    27
    29
    <ned1> ---- needle entry ----
    smal
    <bgn> ==== BEGIN BITMAP NEEDLE ====
      <shar> ---- shared needle ----
      <base> ---- base bitmap name ----
      needle3
    <endo> ==== END BITMAP NEEDLE ====
  <endo> ==== END NEEDLE ====
  <ndl2> ---- bitmap needle ----
  <bgn> ==== BEGIN NEEDLE ====
    <cntr> ---- needle center x,y ----
    27
    29
    <ned1> ---- needle entry ----
    tiny
    <bgn> ==== BEGIN BITMAP NEEDLE ====
      <shar> ---- shared needle ----
      <base> ---- base bitmap name ----
      needle4
    <endo> ==== END BITMAP NEEDLE ====
  <endo> ==== END NEEDLE ====
  <lite> ---- light ----
  plit
  <mask> ---- mask ----
  pnmask20.raw
  <help> ---- help ----
```



```
      Analog Clock  
<endo> ==== END GAUGE ENTRY ====
```

adf_ Generic ADF Radio

Class: CGenericADFRadioGauge

Base Class: CBitmapGauge

Description:

This gauge type implements a generic ADF Radio. The <radi> tag specifies the subsystem ID and unit number of the associated ADF radio. The active and standby frequencies are each displayed as clickable regions, with each digit being independently settable by clicking left/right on the digit. Click area objects for the active frequency (<dga1> through <dga5>), standby frequency (<dgb1> through <dgb5>). The <mode> tag specifies a click area to be used for the ADF radio mode, allowing the user to cycle the radio through the Off, Standby, and On states. The <test> tag specifies a click area to be used for the test function. If the <sdim> tag is included in the gauge specification, then the frequency displays are automatically dimmed when the unit is in Standby mode.

Read Tags

Signature	Description	Type	Format
radi	Radio Identifier	ID_TYPE	ADF radio subsystem ID
		INT	ADF radio subsystem unit number
dga1	Frequency 1 Digit 1	OBJECT	Click Area
dga2	Frequency 1 Digit 2	OBJECT	Click Area
dga3	Frequency 1 Digit 3	OBJECT	Click Area
dga4	Frequency 1 Digit 4	OBJECT	Click Area
dga5	Frequency 1 Digit 5	OBJECT	Click Area
dgb1	Frequency 2 Digit 1	OBJECT	Click Area
dgb2	Frequency 2 Digit 2	OBJECT	Click Area
dgb3	Frequency 2 Digit 3	OBJECT	Click Area
dgb4	Frequency 2 Digit 4	OBJECT	Click Area
dgb5	Frequency 2 Digit 5	OBJECT	Click Area
tune	Tune Button	OBJECT	Click Area
mode	Mode Knob	OBJECT	Knob
stor	Store Button	OBJECT	Click Area
test	Test Button	OBJECT	Click Area
sdim	Standby Dimmed	NO_TYPE	

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
adf_
<bigno> ==== BEGIN OBJECT ====
  <unid> ---- unique id ----
    adfr
  <size> ---- x,y,xsize,ysize ----
    1005
    802
    69
    72
  <radi> ---- radio identifier ----
    adfr
    1
  <dga1> ---- freq 1 digit 1 ----
```



```

<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
        14
        5
        19
        17
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dga2> ---- freq 1 digit 2 ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
        20
        5
        25
        17
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dga3> ---- freq 1 digit 3 ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
        26
        5
        31
        17
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dga4> ---- freq 1 digit 4 ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
        32
        5
        37
        17
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dga5> ---- freq 1 digit 5 ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
        38
        5
        45
        17
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dgb1> ---- freq 2 digit 1 ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
        14
        18
        19
        30
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dgb2> ---- freq 2 digit 2 ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
        20
        18
        25
        30
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dgb3> ---- freq 2 digit 3 ----
<bgn0> ---- begin click area ----

```

```

    <rect> ---- click area rect ----
    26
    18
    31
    30
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dgb4> ---- freq 2 digit 4 ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    32
    18
    37
    30
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<dgb5> ---- freq 2 digit 5 ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    38
    18
    45
    30
    < curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end click area ----
<tune> ---- tune click area ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    46
    53
    63
    70
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Tune Knob
<endo> ---- end click area ----
<mode> ---- mode click area ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    6
    48
    28
    72
    < curs> ---- cursor ----
    mpfinger.csr
    <bmap> ---- bitmap ----
    kaadfmod.pbg
    <help> ---- popup help ----
    Mode Knob
<endo> ---- end click area ----
<stor> ---- Store click area ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    32
    47
    43
    58
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Store Button
<endo> ---- end click area ----
<test> ---- Test click area ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    32
    59

```

```
43
70
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Test Button
<endo> ---- end click area ----
<sdim> ---- standby dimmed ----
<help> ---- popup help ----
ADF Radio
<endo> ==== END OBJECT ====
```

adfg Automatic Direction Finder Needle Gauge

Class: CBasicADFGauge

Base Class: CNeedleGauge

Description:

This gauge type implements an ADF needle gauge. The parent type is a needle gauge, and the <nedl> tag inherited from the base class is used to specify the ADF pointer needle object. The underlying compass plate is defined in the <comp> tag, and the <knob> tag specifies a knob object that can be used to set the compass card heading (does this work?)

Read Tags

Signature	Description	Type	Format
comp	Compass Plate	ID_TYPE	
		OBJECT	Needle
knob	ADF Heading Knob	OBJECT	Knob
hdng	Compass Plate Heading?	OBJECT	Needle

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
adfg
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    adfg
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- radio tag ----
    adfr
  <endo>
  <size> ---- size and location ----
  542
  410
  90
  90
  <over> ---- overlay image ----
  SKYHADFO.PBM
  3
  3
  <cntr> ---- compass center x,y ----
  46
  46
  <rang> ---- angle range ----
  0.0
  360.0
  <nedl> ---- ADF needle ----
  adfn
  <bgno> ==== BEGIN NEEDLE ====
    <file> ---- needle file ----
    adfneed.ndl
  <endo> ==== END NEEDLE ====
  <comp> ---- compass plate ----
  gyro
  <bgno> ==== BEGIN NEEDLE ====
    <file> ---- needle file ----
    DGYRO.NDL
  <endo> ==== END NEEDLE ====
```



```

<knob> ---- ADF heading knob ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhadfk.pbg
    <rect> ---- click area rect ----
    -3
    74
    14
    92
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    ADF Heading
<endo> ---- end click area ----
<help> ---- popup help ----
Auto Direction Finder
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhadfm.raw
<endo> ==== END GAUGE ENTRY ====

```

agrp ADF Radio Group

Class: CADFRadioGroup

Base Class: CGauge

Description:

Unknown. No examples of this gauge type are defined in the default aircraft???

Read Tags

Signature	Description	Type	Format

Example

alti Altimeter Gauge

Class: CAltimeterGauge

Base Class: CBitmapGauge

Description:

This very common gauge type implements a standard analog altimeter gauge. The <mesg> object should specify an altimeter subsystem, usually of type 'ALTI'. The altitude-indicating needles are specified by the <ndl1> (tens), <ndl2> (hundreds) and <ndl3> (thousands) tags. The altimeter gauge also integrates the displays and controls to allow adjustment of the outside barometric pressure (Kollsman setting). The <koll> and <kpxy> tags specify the click area within which the current Kollsman setting is displayed. The <knob> specifies a Knob object with which the user adjusts the Kollsman setting, and the <kmsg> specifies the subsystem object which the gauge exchanges messages about the Kollsman setting, both to GET the current setting for display, and to SET a new value when the user interacts with the Kollsman knob. This will typically be the same 'ALTI' subsystem that is specified in the <mesg> tag, except that the 'baro' datatag is used for the Kollsman setting.

Read Tags

Signature	Description	Type	Format
psiz	Plate Size	INT	X size
		INT	Y size
ndl1	Tens needle	OBJECT	Needle gauge
ndl2	Hundreds needle	OBJECT	Needle gauge
ndl3	Thousands needle	OBJECT	Needle gauge
csrl	Left Cursor	FILE	Cursor (CSR)
csrr	Right Cursor	FILE	Cursor (CSR)
koll	Kollsman Window	OBJECT	Click Area
kpxy	Kollsman Window Offset	INT	X pixels (default 0)
		INT	Y pixels (default 0)
knob	Kollsman Knob	OBJECT	Knob
kmsg	Kollsman Message Data Tag	OBJECT	Message

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
alti
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    alti
    <size> ---- x,y,xsize,ysize ----
    426
    170
    90
    90
    <mesg> -- connect to Altimeter Subsystem --
    <bgno>
        <conn>
        alti
        <dtag>
```



```

        alti
<endo>
<psiz> ---- plate size ----
85
85
<kmsg> ---- Kollsman message ----
<bgn> ---- begin ----
        <conn> ---- group ----
        alti
        <dtag>
        baro
<endo> ---- end ----
<csrl> ---- left side cursor ----
mptleft.csr
<csrr> ---- right side cursor ----
mptright.csr
<ndl1> ---- first needle ----
<bgn> ---- begin needle ----
        <cntr> ---- needle center x,y ----
        46
        46
        <sang> ---- starting angle ----
        0.0
        <dunt> ---- degrees per unit ----
        0.36
        <nedl> ---- bitmap needle ----
        stnd
        <bgn> ===== BEGIN BITMAP NEEDLE =====
                <file> ---- needle file ----
                needle1.ndl
        <endo> ===== END BITMAP NEEDLE =====
<endo> ---- end needle ----
<ndl2> ---- second needle ----
<bgn> ---- begin needle ----
        <cntr> ---- needle center x,y ----
        46
        46
        <sang> ---- starting angle ----
        0.0
        <dunt> ---- degrees per unit ----
        0.036
        <nedl> ---- bitmap needle ----
        stub
        <bgn> ===== BEGIN BITMAP NEEDLE =====
                <file> ---- needle file ----
                needle2.ndl
        <endo> ===== END BITMAP NEEDLE =====
<endo> ---- end needle ----
<ndl3> ---- third needle ----
<bgn> ---- begin needle ----
        <cntr> ---- needle center x,y ----
        46
        46
        <sang> ---- starting angle ----
        0.0
        <dunt> ---- degrees per unit ----
        36.0
        <nedl> ---- bitmap needle ----
        tenk
        <bgn> ===== BEGIN BITMAP NEEDLE =====
                <file> ---- needle file ----
                ALTI10K.NDL
        <endo> ===== END BITMAP NEEDLE =====
<endo> ---- end needle ----
<help> ---- popup help ----
Altimeter
<koll> ---- Kollsman window ----
<bgn> ---- begin click area ----
        <rect> ---- click area rect ----
        69
        34

```

```
88
55
<endo> ---- end click area ----
<knob> ---- Kollsman knob ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhknob.pbg
    <rect> ---- click area rect ----
    -2
    74
    15
    92
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Kollsman Knob
<endo> ---- end click area ----
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhaltm.raw
<endo> ==== END GAUGE ENTRY ====
```

altr Rolling Altimeter Gauge

Class: CRollingAltimeterGauge

Base Class: CBitmapGauge

Description:

This gauge type is another type of analog altimeter gauge. Instead of using needles for each of the units, it displays the current altitude using simulated rolling wheels (like a car odometer).

The artwork for the digits is supplied via the <zdig>, <ndig> and <tdig> tags, each of which provide the filename of a PBM artwork file. These PBM files contain all digits 0-9, with the individual sub-images stacked vertically. See the example PBM files to see how the digits need to be stacked in each file. The positions of the digits on the gauge is specified by the <digx> and <digy> fields. The values of these parameters will be determined by the overall size of the gauge, and the sizes of the individual digits in the digit strip artwork files as specified in the <dsiz> tag.

A single needle is defined using the <ndl1> tag. This needle rotates once per thousand feet, and allows for faster visual detection of altitude change than having to decipher the digital display. Like the 'alti' gauge type, the Rolling Altimeter also has a Kollsman display window and knob, specified using the same <koll>, <kpxy>, <knob> and <kmsg> data tags.

Read Tags

Signature	Description	Type	Format
psiz	Plate Size	INT	X size
		INT	Y size
zdig	Zero Digits Strip	ART	PBM filename
ndig	Number Digits Strip	ART	PBM filename
tdig	Ten-Thousands Digit Strip	OBJECT	Needle gauge
digx	Digits X-position	INT	X pixel position (1 st)
		INT	X pixel position (2 nd)
		INT	X pixel position (3 rd)
		INT	X pixel position (4 th)
digy	Digits Y-position	INT	Y pixel position
dsiz	Digits Size in Strips	INT	X pixel size
		INT	Y pixel size
ndl1	Feet Needle	OBJECT	Needle
koll	Kollsman Window	OBJECT	Click Area
kpxy	Kollsman Window Offset	INT	X pixels (default 0)
		INT	Y pixels (default 0)
knob	Kollsman Knob	OBJECT	Knob
kmsg	Kollsman Message Data Tag	OBJECT	Message

Example (TRI30.PNL)

<gage> ==== GAUGE ENTRY ====



```

altr
<bgn> ===== BEGIN GAUGE ENTRY =====
    <unid> ---- unique id ----
    alti
    <size> ---- x,y,xsize,ysize ----
    421
    530
    90
    90
    <mesg>
    <bgn>
        <conn>
        Alt1
        <dtag>
        alti
    <endo>
    <psiz> ---- plate size ----
    85
    85
    <conn> ---- data connection ----
    alti
    <kmsg> ---- Kollsman message ----
    <bgn> ---- begin ----
        <conn> ---- group ----
        Alt1
        <dtag> ---- user data ----
        baro
    <endo> ---- end ----
    <zdig> ---- zero digits strip ----
    kazstrip.pbm
    <ndig> ---- number digits strip ----
    kadstrip.pbm
    <tdig> ---- ten thousands digits strip ----
    katstrip.pbm
    <digx> ---- X offsets for each digit column (4) ----
    26
    34
    46
    52
    58
    <digy> ---- Y offset for all digits ----
    29
    <dsiz> ---- digit size X,Y in strips ----
    6
    11
    <ndl1> ---- first needle ----
    <bgn> ---- begin needle ----
        <cntr> ---- needle center x,y ----
        45
        47
        <sang> ---- starting angle ----
        0.0
        <dunt> ---- degrees per unit ----
        0.36
        <nedl> ---- bitmap needle ----
        ndl7
        <bgn> ===== BEGIN BITMAP NEEDLE =====
            <file> ---- needle file ----
            needle7.ndl
        <endo> ===== END BITMAP NEEDLE =====
    <endo> ---- end needle ----
    <help> ---- popup help ----
    Altimeter
    <koll> ---- Kollsman window ----
    <bgn> ---- begin click area ----
        <rect> ---- click area rect ----
        36
        56
        55
        62
    <endo> ---- end click area ----

```

```
<knob> ---- Kollsman knob ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhknob.pbg
    <rect> ---- click area rect ----
    3
    69
    20
    87
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Kollsman Knob
<endo> ---- end click area ----
<lite> ---- light flag ----
pfl1
<mask> ---- light mask ----
kamask02.raw
<endo> ==== END GAUGE ENTRY ====
```

amms Ammeter Needle Gauge

Class: CAmmeterGauge

Base Class: CNeedleGauge

Description:

This gauge is a specialized needle gauge for displaying electrical current. It does not appear to have any additional tags or any different functionality from a standard 'need' type Needle Gauge.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
amms
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    amms
    <over> ---- overlay image ----
    skyhamp.pbm
    0
    0
    <size> ---- placement and size ----
    142
    375
    33
    66
    <cntr> ---- needle center x,y ----
    19
    32
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- info tag ----
        chrg
        <user> ---- user data ----
        HARDWARE,GAUGE
        <dtag>
        indn
    <endo>
    <nedl> ---- bitmap needle ----
    smal
    <bgno> ==== BEGIN BITMAP NEEDLE ====
        <file> ---- needle file ----
        needle3.ndl
    <endo> ==== END BITMAP NEEDLE ====
    <gtbl> ---- gauge value table ----
    <bgno> ---- begin table ----
        <fmt1> - x y
        -60.0 216.7
        -30.0 243.1
        0.0 270.0
        30.0 295.6
        60.0 321.2
        <endf> - end format
    <endo> ---- end table ----
    <gmin> ---- min value allowed ----
    -60.0
    <gmax> ---- max value allowed ----
    60.0
```



```
<help> ---- popup help ----  
Ammeter Indicator  
<lite> ---- light flag ----  
gage  
<mask> ---- light mask ----  
skyhampm.raw  
<endo> ==== END GAUGE ENTRY ====
```

annc Annunciator Light

Class: CAnnunciatorLightGauge

Base Class: CGauge

Description:

This very common indicator gauge represents an annunciator light. It has only two states, on and off, which it determines by polling the subsystem specified in the <mesg> data tag. The artwork for an annunciator must be a two-frame PBG image file. Frame 0 is displayed if the subsystem is in the Off state, and frame 1 is displayed if the subsystem is in the On state. Typically, the artwork would consist of an indicator light graphic "unlit" in frame 0 and "lit" in frame 1.

Read Tags

Signature	Description	Type	Format
bmap	Annunciator artwork	ART	PBG filename

Example (TRI200.PNL)

This sample annunciator is just the top (nosegear) indicator of the three gear indicator lights of the TRI Kodiak twin piston-engine aircraft.

```
<gage> ==== GAUGE ENTRY ====
annc
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  gdna
  <size> ---- x,y,xsize,ysize ----
  404
  511
  19
  20
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- connected to ----
    gNGD
    <user> ---- user data ----
    HARDWARE,LIGHT
    <type>
    INT,0
  <endo>
  <bmap> ---- bitmap ----
  pngearn1.pbg
  <lite> ---- light ----
  anlt
  <help> ---- popup help ----
  Nose Gear Down Annunciator
<endo> ==== END GAUGE ENTRY ====
```



aoan Angle of Attack Dual Needle Gauge

Class: CAOANeedleGauge

Base Class: CTwoNeedleGauge

Description:

This specialized dual needle gauge is intended to show the aircraft angle of attack. The speed ratio is indicated by the needle specified by the <ndl1> tag, and modified by <knob>. The angle of attack needle is specified by the <ndl2> tag.

Read Tags

Signature	Description	Type	Format
knob	Speed Ratio Knob	OBJECT	Knob

Example (TRI40.PNL)

```
<gage> ===== GAUGE ENTRY =====
aoan
<bgn> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
  aoai
  <size> ---- placement and size ----
  145
  515
  77
  77
  <mesg> ---- message ----
  <bgn> ---- begin ----
    <conn> ---- info tag ----
    fosc
    <dtag> ---- data tag ----
    spdR
  <endo> ---- end ----
  <ndl1> ---- engine 1 manifold needle ----
  <bgn> ---- BEGIN NEEDLE ----
    <mesg> ---- message ----
    <bgn> ---- begin ----
      <conn> ---- info tag ----
      fosc
      <dtag> ---- data tag ----
      spdR
    <endo> ---- end ----
    <cntr> ---- needle center x,y ----
    38
    38
    <ned1> ---- bitmap needle ----
    aoai
    <bgn> ===== BEGIN BITMAP NEEDLE =====
      <file> ---- needle file ----
      rhaoaind.ndl
    <endo> ===== END BITMAP NEEDLE =====
    <gtbl> ---- gauge value table ----
    <bgn> ---- begin table ----
      <fmt1> - x y
      1.2 0.0
      1.3 23.2
      1.4 44.0
      1.5 62.7
      <endf> - end format
    <endo> ---- end table ----
```



```

        <sang> ---- starting angle ----
        226.0
        <gmin> ---- min value allowed ----
        1.2
        <gmax> ---- max value allowed ----
        1.5
    <endo> ---- END NEEDLE ----
    <ndl2> ---- engine 2 manifold needle ----
    <bgno> ---- BEGIN NEEDLE ----
        <mesg> ---- message ----
        <bgno> ---- begin ----
            <conn> ---- info tag ----
            f0sm
            <dtag> ---- data tag ----
            lftR
        <endo> ---- end ----
        <cntr> ---- needle center x,y ----
        38
        38
        <nedl> ---- bitmap needle ----
        kan1
        <bgno> ===== BEGIN BITMAP NEEDLE =====
            <file> ---- needle ----
            kaneed1.ndl
        <endo> ===== END BITMAP NEEDLE =====
        <gtbl> ---- gauge value table ----
        <bgno> ---- begin table ----
            <fmt1> - x y
            0.0 202.0
            0.2 180.0
            0.4 135.0
            0.6 90.0
            0.8 45.0
            1.0 0.0
            <endf> - end format
        <endo> ---- end table ----
        <sang> ---- starting angle ----
        0.0
        <gmin> ---- min value allowed ----
        0.0
        <gmax> ---- max value allowed ----
        1.0
    <endo> ---- END NEEDLE ----
    <knob> ---- speed ratio knob ----
    <bgno> ---- begin ----
        <rect> ---- rect ----
        0
        62
        14
        77
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Speed Ratio
    <endo> ---- end ----
    <lite> ---- light ----
    plf1
    <mask> ---- light mask ----
    rhmask01.raw
    <help> ---- popup help ----
    Angle of Attack (AOA)
    <endo> ===== END GAUGE ENTRY =====

```

apre Analog Altitude Preselect Gauge

Class: CAnalogAltPreselectGauge

Base Class: CGauge

Description:

The Analog Altitude Preselect Gauge allows the autopilot altitude preset to be selected. The <mesg> tag must specify the aircraft's autopilot subsystem of type 'AXIS'.

The <knob> tag specifies a click area and graphics for the altitude selection knob; left-clicking on this knob decreases the altitude, right-clicking on the knob increases it.

The preselected altitude value is displayed using the digit strip artwork specified in the <fdig>, <digi>, and <zero> tags. These digit strips contain individual sub-images for all digits 0-9. See the example artwork to see how the sub-images are stacked within the PBM files.

The <aler> tag specifies an annunciator area that is Off except when the autopilot is approaching (within 1000 feet) the preset altitude.

Read Tags

Signature	Description	Type	Format
knob	Altitude Select Knob	OBJECT	Knob
fdig	FL Digits Strip	ART	PBM filename
digi	Digits Strip	ART	PBM filename
zero	Zero Digits Strip	ART	PBM filename
thou	Thousands Digit Offset	INT	X pixel offset
		INT	Y pixel offset
hund	Hundreds Digit Offset	INT	X pixel offset
		INT	Y pixel offset
ones	Ones Digit Offset	INT	X pixel offset
		INT	Y pixel offset
dspc	Digits Spacing	INT	Pixels
zspc	Zero Spacing	INT	Pixels
aler	Altitude Alert Annunciator	OBJECT	Click Area

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
apre
<bgn0> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  apre
  <mesg> ---- Subsystem Message ----
  <bgn0>
    <conn> ---- autopilot ----
    AXIS
    <user> ---- user data ----
    HARDWARE, OTHER
    <dtag> ---- user data ----
    _FCS
```



```

<endo>
<size> ---- x,y,xsize,ysize ----
833
582
101
41
<aler> ---- Alt Alert ----
<bgn> ---- begin ----
    <rect> ---- rect ----
        7
        10
        28
        29
    <bmap> ---- bitmap ----
    kaaalert.pbg
    <help> ---- help ----
    Altitude Alert
    <curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end ----
<knob> ---- Knob ----
<bgn> ---- begin ----
    <rect> ---- rect ----
        75
        29
        83
        37
    <help> ---- help ----
    Altitude Knob
    <curs> ---- cursor ----
    mpfinger.csr
<endo> ---- end ----
<fdig> ---- FL digit strip ----
katstrip.pbm
<digi> ---- digit strip ----
kadstrip.pbm
<zero> ---- zero strip ----
kazstrip.pbm
<thou> ---- thousands offset -----
41
15
<hund> ---- hundreds offset ----
61
15
<ones> ---- ones offset ----
71
15
<dspc> ---- digit spacing ----
3
<zspc> ---- zero spacing ----
2
<help> ---- popup help ----
Altitude Preselect
<endo> ===== END GAUGE ENTRY =====

```

aspd Airspeed Indicator Gauge

Class: CAirspeedGauge

Base Class: CNeedleGauge

Description:

This common gauge type is a standard airspeed indicator. It is descended from the generic needle gauge type 'need' so most of the data tags are inherited. The <nedl> tag specifies the airspeed needle.

Gauge-specific tags include a <knob> which controls the TAS adjustment, and the MMO needle which indicates ???

Read Tags

Signature	Description	Type	Format
knob	TAS Adjust Knob	OBJECT	Knob
mmo_	MMO Needle	OBJECT	Needle

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
aspd
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  aspd
  <size> ---- x,y,xsize,ysize ----
  197
  170
  90
  90
  <cntr> ---- needle center x,y ----
  45
  45
  <mesg> -- connect to Airspeed Subsystem --
  <bgno>
    <conn>
    sped
    <dtag>
    sped
  <endo>
  <rang> ---- angle range ----
  0.0
  337.0
  <gmin> ---- min value allowed ----
  0.0
  <gmax> ---- max value allowed ----
  200.0
  <nedl> ---- bitmap needle ----
  stnd
  <bgno> ==== BEGIN BITMAP NEEDLE ====
    <file> ---- needle definition ----
    needle1.ndl
  <endo> ==== END BITMAP NEEDLE ====
  <gtbl> ---- gauge value table ----
  <bgno> ---- begin table ----
    <fmt1> - x y
    0.0 0.0
    40.0 21.0
    45.0 34.5
    50.0 46.7
    55.0 58.3
```



```

60.0 70.6
65.0 80.0
70.0 90.9
75.0 100.4
80.0 110.6
85.0 121.3
90.0 133.3
95.0 144.5
100.0 156.7
105.0 168.6
110.0 180.0
115.0 192.2
120.0 204.7
125.0 216.5
130.0 227.5
135.0 234.5
140.0 242.2
145.0 249.1
150.0 255.7
155.0 263.0
160.0 270.0
165.0 277.0
170.0 284.0
175.0 292.7
180.0 301.4
185.0 309.9
190.0 318.9
195.0 327.8
200.0 336.9
<endf> - end format
<endo> ---- end table ----
<knob> ---- true airspeed knob ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhknob.pbg
    <rect> ---- click area rect ----
    -3
    75
    14
    93
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    True Airspeed Adjustment
<endo> ---- end click area ----
<help> ---- help string ----
Airspeed Indicator
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhaspm.raw
<endo> ===== END GAUGE ENTRY =====

```

atop **Basic King Autopilot Gauge**

Class: CBasicKingAutopilotGauge

Base Class: CBitmapGauge

Description:

Unknown. No examples in the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

atti Attitude Indicator Gauge

Class: CAttitudeGauge

Base Class: CBitmapGauge

Description:

Unknown. No examples in the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

batt Basic Battery Switch

Class: CBasicBatterySwitch

Base Class: CSimpleSwitchGauge

Description:

This is a simple switch intended to represent the Battery Master switch.

Read Tags

Signature	Description	Type	Format
altt	Alternator Datatag	ID_TYPE	
batt	Battery Datatag	ID_TYPE	

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
batt
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    batt
    <mesg>
    <bgno>
        <conn>
        mBat
        <dtag>
        st8t
    <endo>
    <altt> ---- alt data tag ----
    mAlt
    <batt> ---- battery data tag ----
    mBat
    <size> ---- switch size ----
    109
    576
    36
    50
    <bmap> ---- bitmap name ----
    skyhpowr.pbg
    <csru> ---- up cursor ----
    mpflipup.csr
    <csrd> ---- down cursor ----
    mpflipdn.csr
    <sfxu> ---- up sfx ----
    switchup.wav
    <sfxd> ---- down sfx ----
    switchdn.wav
    <help> ---- popup help ----
    Alternator/Battery Switch
<endo> ==== END GAUGE ENTRY ====
```



bdcg Basic Digital Clock Gauge

Class: CBasicDigitalClockGauge

Base Class: CFlyhawkDigitalClockGauge

Description:

This gauge type implements a digital clock with some differences from the 'cdcg' Flyhawk digital clock. Most data tags are inherited from that gauge type, however.

In addition to the capabilities of the Flyhawk Digital Clock, this gauge type supports the display of either universal (UTC) time, local time, flight time or elapsed time. Gauge-specific data tags <tick>, <tkut>, <tklt>, <tkft> and <tktet> control the display of the "tick" that indicates which time is being displayed.

Read Tags

Signature	Description	Type	Format
tick	Tick Offset	INT	X pixel offset
		INT	Y pixel offset
tkut	UTC Tick	INT	X pixel offset
tklt	Local Time Tick	INT	X pixel offset
tkft	Flight Time Tick	INT	X pixel offset
tktet	Elapsed Time Tick	INT	X pixel offset

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
bdcg
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    pdcg
    <size> ---- x,y,xsize,ysize ----
      92
      430
      67
      67
    <clkd> ---- clock display area ----
      16
      20
      51
      32
    <mesg> ---- Subsystem Message ----
    <bgno>
      <conn> ---- data connection ----
        pdcg
        <user> ---- user data ----
          HARDWARE, GAUGE
        <user> ---- user data ----
          UNIT,1
        <dtag> ---- user data ----
          lmod
    <endo>
    <bset> ---- select button ----
    <bgno> ---- begin button ----
      <rect> ---- rectangle ----
        13
        48
        27
```



```

55
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Select Button
<endo> ---- end button
<bcon> ---- control button ----
<bgno> ---- begin button ----
<rect> ---- rectangle ----
39
48
54
55
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Control Button
<endo> ---- end button
<bbot> ---- both buttons ----
<bgno> ---- begin button ----
<rect> ---- rectangle ----
29
48
37
57
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Set Mode (Select/Control)
<endo> ---- end button
<tick> ---- tick offset ----
34
2
<tkut> ---- UT/GMT tick ----
21
<tklt> ---- LT tick ----
33
<tkft> ---- FT tick ----
41
<tket> ---- ET tick ----
50
<ft13> ---- large font ----
<frgb> ---- font color ----
180
0
0
<noco> ---- no colon ----
<help> ---- popup help ----
Chronometer
<endo> ===== END GAUGE ENTRY =====

```

bgge Generic Bitmap Gauge

Class: CBitmapGauge

Base Class: CGauge

Description:

The generic bitmap gauge type is not usually instantiated, however many other gauge types are derived from it. This gauge class defines data tags which allow the gauge to load a PBM bitmap via the <bmap> tag and an overlay bitmap via the <over> tag. When the gauge is displayed, the <bmap> image is displayed as the base or "lowest" layer, prior to any other needles, digits, etc. that the gauge might display. The <over> image is displayed as the last or "highest" layer, after all other gauge-specific information. A typical example of the <over> tag is the center bezel that covers the "hub" of a typical needle gauge.

Read Tags

Signature	Description	Type	Format
bmap	Bitmap image	ART	PBM filename
		INT	X pixel size
		INT	Y pixel size
over	Overlay image	ART	PBM filename
		INT	X pixel size
		INT	Y pixel size

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
bgge
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    pmod
    <size> ---- x,y,xsize,ysize ----
    564
    145
    172
    68
    <lite> ---- light ----
    plcp
    <mask> ---- light mask ----
    rhctrmp.raw
    <nomc> ---- no mouse clicks ----
<endo> ==== END GAUGE ENTRY ====
```

brpg Brake Pressure Gauge

Class: CBrakePressureGauge

Base Class: CTwoNeedleGauge

Description:

This gauge type is a specialized dual needle gauge intended to display hydraulic brake pressure. There do not appear to be any gauge-specific data tags, but the example does not include any <mesg> subsystem specifications, so there may be some specific implementation that gets the brake pressure value from some hard-coded subsystem.

Read Tags

Signature	Description	Type	Format

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
brpg
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    brpg
  <size> ---- x,y,xsize,ysize ----
    835
    533
    55
    55
  <ndl1> ---- Left Pressure ----
    <cntr> ---- needle center x,y ----
      6
      28
    <gmin> ---- min value allowed ----
      0.0
    <gmax> ---- max value allowed ----
      2000.0
    <nedl> ---- bitmap needle ----
      rhn3
    <bgn> ==== BEGIN BITMAP NEEDLE ====
      <file> ---- needle file ----
        rhneedl3.ndl
    <endo> ==== END BITMAP NEEDLE ====
    <sang> ---- starting angle ----
      19.5
    <gtbl> ---- gauge value table ----
    <bgn> ---- begin table ----
      <fmt1> - x y
        0.0 86.5
        1000.0 42.5
        2000.0 0.0
      <endf> - end format
    <endo> ---- end table ----
  <endo> ---- end needle ----
  <ndl2> ---- Right Pressure ----
    <cntr> ---- needle center x,y ----
      48
      28
    <gmin> ---- min value allowed ----
      0.0
    <gmax> ---- max value allowed ----
      2000.0
```



```

<nedl> ---- bitmap needle ----
rhn3
<bgno> ===== BEGIN BITMAP NEEDLE =====
    <file> ---- needle file ----
        rhneed13.ndl
<endo> ===== END BITMAP NEEDLE =====
<sang> ---- starting angle ----
254.0
<gtbl> ---- gauge value table ----
<bgno> ---- begin table ----
    <fmt1> - x y
        0.0 0.0
        1000.0 44.0
        2000.0 86.0
    <endf> - end format
<endo> ---- end table ----
<endo> ---- end needle ----
<left> ---- left help area ----
<bgno> ---- begin ----
    <rect> ----
        0
        7
        26
        35
    <help> ----
        Brake Pressure (Left)
<endo> ---- end ----
<right> ---- right help area ----
<bgno> ---- begin ----
    <rect> ----
        28
        7
        54
        35
    <help> ----
        Brake Pressure (Right)
<endo> ---- end ----
<lite> ---- light ----
plfl
<mask> ---- light mask ----
rhmask13.raw
<help> ---- help string ----
    Brake Pressure
<endo> ===== END GAUGE ENTRY =====

```

cadd Collins Air Data Gauge

Class: CCollinsADGGauge

Base Class: CBitmapGauge

Description:

This gauge type implements the Collins Air Data gauge, which displays the aircraft altitude, Kollsman setting, vertical speed, and autopilot vertical speed bug. The <mesg> tag should specify the air data subsystem of type 'padd'.

The <scrn> tag specifies the location of the display screen relative to the top-left corner of the gauge region. There is no way to alter the font sizes or display styles of the gauge; these are all hard-coded within the gauge implementation.

Several click areas are defined which allow user interaction with the air data system. The <baro> click area allows the user to change the outside barometric pressure (Kollsman setting). The <in__> and <mb__> click areas switch the Kollsman setting between inches and millibars, respectively. The <alt__> click area controls the autopilot altitude preselect, which is displayed at the bottom-right corner of the display screen. The <vs__> click area controls the autopilot vertical speed reference, which is displayed as a magenta bug on the vertical speed display arc.

Read Tags

Signature	Description	Type	Format
scrn	Display Screen	INT	Top-Left X position
		INT	Top-Left Y position
		INT	Bottom-Right X position
		INT	Bottom-Right Y position
baro	Barometric Setting Knob	OBJECT	Knob
vs__	VS Reference Knob	OBJECT	Click Area
alt__	Altitude Preselect Knob	OBJECT	Click Area
in__	Inches Button	OBJECT	Click Area
mb__	Millibars Button	OBJECT	Click Area

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
cadd
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    padd
    <mesg>
    <bgno>
        <conn> ---- data tag ----
        padd
    <endo>
    <size> ---- x,y,xsize,ysize ----
    419
    419
    102
```



```

131
<scrn> ---- screen ----
<bgn> ---- begin ----
    <rect> ---- rect ----
        4
        5
        97
        94
    <endo> ---- end ----
<baro> ---- baro knob ----
<bgn> ---- begin ----
    <rect> ---- rect ----
        3
        106
        24
        129
    <curs> ---- cursor ----
    mpfinger.csr
    <bmap> ---- bitmap ----
    rhadinmb.pbg
    <help> ---- help ----
    Barometer
<endo> ---- end ----
<vs__> ---- VS ref knob ----
<bgn> ---- begin ----
    <rect> ---- rect ----
        42
        110
        59
        128
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Vertical Speed
<endo> ---- end ----
<alt_> ---- Altitude knob ----
<bgn> ---- begin ----
    <rect> ---- rect ----
        80
        110
        97
        128
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Altitude
<endo> ---- end ----
<in__> ---- Inches ----
<bgn> ---- begin ----
    <rect> ---- rect ----
        0
        95
        12
        105
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Inches
<endo> ---- end ----
<mb__> ---- Millibars ----
<bgn> ---- begin ----
    <rect> ---- rect ----
        13
        95
        27
        105
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Millibars
<endo> ---- end ----

```

```
<lite> ---- light ----  
plef  
<help> ---- popup help ----  
Air Data System  
<endo> ==== END GAUGE ENTRY ====
```

cbtn Covered Light Button

Class: CCoveredLightButtonGauge

Base Class: CCoveredSwitch

Read Tags

Signature	Description	Type	Format

Example (TRI45.PNL)

```
<gage> ==== GAUGE ENTRY ====
cbtn
<bgn> ==== BEGIN GAUGE ENTRY ====
  <size> ---- x,y,xsize,ysize ----
    325
    518
    61
    77
  <mesg> ---- button message ----
  <bgn> ---- begin ----
    <conn> ---- connection ----
    Afir
    <user> ---- hardware ----
    HARDWARE, SWITCH
  <endo> ---- end ----
  <lmsg> ---- light message ----
  <bgn> ---- begin ----
    <conn> ---- connection ----
    APUF
    <user> ---- hardware ----
    HARDWARE, LIGHT
  <endo> ---- end ----
  <bmap> ---- bitmap ----
  rhapuf1.pbg
  <lbmp> ---- lit bitmap ----
  rhapuf2.pbg
  <mmnt> ---- momentary ----
  <curs> ---- cursor ----
  mpfinger.csr
  <csru> ---- up cursor ----
  mpflipdn.csr
  <csrd> ---- down cursor ----
  mpflipup.csr
  <sfxu> ---- up sfx ----
  switchup.wav
  <sfxd> ---- down sfx ----
  switchdn.wav
  <help> ---- popup help ----
  APU Fire Test
<endo> ==== END GAUGE ENTRY ====
```



cdcg Flyhawk Digital Clock Gauge

Class: CFlyhawkDigitalClockGauge

Base Class: CBitmapGauge

Description:

Use this gauge type for the Davtron style digital clock, temperature and voltmeter gauge. Must Connect to the Digital Clock OAT subsystem, 'doat'.

Read Tags:

Signature	Description	Type	Format
oatd	Outside air temperature display	INT	Top-left X location
		INT	Top-left Y location
		INT	Bottom-right X location
		INT	Bottom-right Y location
clkd	Clock display	INT	Top-left X location
		INT	Top-left Y location
		INT	Bottom-right X location
		INT	Bottom-right Y location
bupp	Upper Button	OBJECT	Click Area
bsel	Select Button	OBJECT	Click Area
bcon	Control Button	OBJECT	Click Area
bbot	Both Buttons (typically placed between Select and Control)	OBJECT	Click Area
ft13	Digital 13-point font (specify this tag if you need a larger font than the default)	NO_VALUE	
noco	No Colon (use this if base image already has the colon drawn on it)	NO_VALUE	
cgap	Colon Gap (use this to specify the gap you would like to leave for the colon)	INT	Pixels
frgb	Font RGB Use this to specify the colour for the digital readout	INT	R component
		INT	G component
		INT	B component

Example (FLYHWK00.PNL)

<gage> ==== GAUGE ENTRY ====



```

cdcg
<bgno> ===== BEGIN GAUGE ENTRY =====
<unid> ---- unique id ----
cdcg
<size> ---- x,y,xsize,ysize ----
116
199
51
61
<oatd> ---- O.A.T. display area ----
5
16
42
24
<clkd> ---- clock display area ----
2
27
46
42
<mesg> ---- Subsystem Message ----
<bgno>
    <conn> ---- data connection ----
    doat
    <user> ---- user data ----
    HARDWARE, GAUGE
    <user> ---- user data ----
    UNIT,1
    <dtag> ---- user data ----
    umod
<endo>
<bupp> ---- upper button ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    22
    1
    28
    7
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Upper Button
<endo> ---- end button
<bsel> ---- select button ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    13
    52
    19
    57
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Select Button
<endo> ---- end button
<bcon> ---- control button ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    32
    52
    38
    57
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Control Button
<endo> ---- end button
<bbot> ---- both buttons ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    20
    52

```

```
31
57
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Set Mode (Select/Control)
<endo> ---- end button
<help> ---- popup help ----
Digital Clock/O.A.T.
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhoatm.raw
<endo> ==== END GAUGE ENTRY ====
```

cesp Collins EFIS Selection Panel

Class: CCollinSEFISSelectionPanel

Base Class: CBitmapGauge

Description:

This gauge type implements the EFIS mode selection panel for the Collins EFIS avionics suite equipped in the TRI Peregrine 800TR jet.

Read Tags

Signature	Description	Type	Format
brg_	Bearing	OBJECT	Click Area
brgm	Bearing Mode	OBJECT	Click Area
brgk	Bearing Knob	OBJECT	Knob
DH_	Decision Height Knob	OBJECT	Knob
form	Display Format	OBJECT	Click Area
crs_	Course	OBJECT	Click Area
crsm	Course Mode	OBJECT	Click Area
crsk	Course Knob	OBJECT	Knob
wx_	Weather Display	OBJECT	Click Area
nav_	Nav Data Toggle	OBJECT	Click Area
pfdm	PFD Message	OBJECT	Message
ehim	EHSI Message	OBJECT	Message
mndm	Multi-ND Message	OBJECT	Message
adm_	Air Data Message	OBJECT	Message
aplt	Autopilot Message	OBJECT	Message

Example (TRI40.PNL)

```

<gage> ==== GAUGE ENTRY ====
cesp
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  ppsp
  <size> ---- x,y,xsize,ysize ----
  575
  143
  152
  68
  <brg_> ---- Bearing ----
  <bgn> ---- begin ----
    <rect> ---- rect ----
    16
    41
    28
    52
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Bearing
  <endo> ---- end ----
  <brgm> ---- BRG mode ----
  <bgn> ---- begin ----
    <rect> ---- rect ----
    2
    22
    45
    59
    <curs> ---- cursor ----

```



```

        mpfinger.csr
        <help> ---- help ----
        Bearing Source
<endo> ---- end ----
<brgk> ---- BRG knob ----
<bgn0> ---- begin ----
        <rect> ---- rect ----
        7
        32
        36
        64
        <bmap> ---- bitmap ----
        rhpbrg.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Bearing Knob
<endo> ---- end ----
<dh__> ---- DH knob ----
<bgn0> ---- begin ----
        <rect> ---- rect ----
        35
        10
        49
        24
        <bmap> ---- bitmap ----
        rhndtest.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Decision Height
<endo> ---- end ----
<form> ---- Display Format ----
<bgn0> ---- begin ----
        <rect> ---- rect ----
        59
        14
        82
        38
        <bmap> ---- bitmap ----
        rhpfmt.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Display Format
<endo> ---- end ----
<crs_> ---- Course Source ----
<bgn0> ---- begin ----
        <rect> ---- rect ----
        118
        42
        130
        53
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Course
<endo> ---- end ----
<crsm> ---- Course Source ----
<bgn0> ---- begin ----
        <rect> ---- rect ----
        103
        24
        145
        60
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Course Source
<endo> ---- end ----
<crsk> ---- Course Knob ----

```

```

<bgn0> ---- begin ----
    <rect> ---- rect ----
    109
    32
    138
    64
    <bmap> ---- bitmap ----
    rhpcrs.pbg
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Course
<endo> ---- end ----
<wx_> ---- WX Display ----
<bgn0> ---- begin ----
    <rect> ---- rect ----
    63
    45
    75
    57
    <bmap> ---- bitmap ----
    rhndwx.pbg
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Weather Radar Toggle
<endo> ---- end ----
<nav_> ---- Nav Data ----
<bgn0> ---- begin ----
    <rect> ---- rect ----
    97
    11
    109
    23
    <bmap> ---- bitmap ----
    rhndnav.pbg
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Nav Data Toggle
<endo> ---- end ----
<pfdm> ---- PFD message ----
<bgn0> ---- begin ----
    <conn> ---- group ----
    ppfd
    <dtag>
    nsys
<endo> ---- end ----
<ehim> ---- EHI message ----
<bgn0> ---- begin ----
    <conn> ---- group ----
    pnvd
<endo> ---- end ----
<mndm> ---- MND message ----
<bgn0> ---- begin ----
    <conn> ---- group ----
    pmnd
<endo> ---- end ----
<adm_> ---- ADM message ----
<bgn0> ---- begin ----
    <conn> ---- group ----
    padd
<endo> ---- end ----
<aplt> ---- AUTOPILOT message ----
<bgn0> ---- begin ----
    <conn> ---- group ----
    pilo
<endo> ---- end ----
<help> ---- help ----
    EFIS Mode Selector
<endo> ==== END GAUGE ENTRY ====

```


cgrp Comm Radio Group

Class: CCommRadioGroup

Base Class: CGauge

Description:
Unknown. No examples in default aircraft.

Read Tags

Signature	Description	Type	Format

Example

cmnd Collins Multi-Nav Display

Class: CCollinsMNDGauge

Base Class: CBitmapGauge

Description:

This gauge is part of the Collins EFIS avionics suite equipped in the TRI Peregrine 800TR jet.

Read Tags

Signature	Description	Type	Format
obs_	OBS Message	Object	Message
ap__	Autopilot Message	Object	Message
frmt	Display Format	Object	Click Area
hdg_	Heading Knob	Object	Click Area
crs_	Course Knob	Object	Click Area
brgt	Brightness Knob	Object	Click Area

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
cmnd
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    pmnd
    <mesg>
    <bgno>
      <conn> ---- data tag ----
        pmnd
      <endo>
      <size> ---- x,y,xsize,ysize ----
        424
        562
        96
        96
      <over> ---- overlay ----
        rhpmndo.pbm
        0
        0
      <obs_> --- OBS ----
      <bgno> --- begin ----
        <conn>
          'mno1'
        <user>
          HARDWARE, GAUGE
      <endo> ---- end ----
      <ap__> ---- autopilot ----
      <bgno> ---- begin ----
        <conn>
          'pilo'
        <user>
          HARDWARE, OTHER
      <endo> ---- end ----
      <frmt> ---- format ----
      <bgno> ---- begin ----
        <rect> ---- rect ----
          81
          3
          96
          18
        <curs> ---- cursor ----
```



```

        mpfinger.csr
        <help> ---- help ----
        Format
    <endo> ---- end ----
    <hdg_> ---- heading ----
    <bgno> ---- begin ----
        <rect> ---- rect ----
        0
        79
        11
        96
        < curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Heading
    <endo> ---- end ----
    <crs_> ---- course ----
    <bgno> ---- begin ----
        <rect> ---- rect ----
        81
        79
        96
        96
        < curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Course
    <endo> ---- end ----
    <brgt> ---- brightness ----
    <bgno> ---- begin ----
        <rect> ---- rect ----
        0
        0
        10
        17
        < curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Brightness
    <endo> ---- end ----
    <help> ---- popup help ----
    Multi Navigation Display
    <endo> ===== END GAUGE ENTRY =====

```

cnvd Collins Navigational Display

Class: CCollinsNDGauge

Base Class: CBitmapGauge

Description:

This gauge implements the Collins navigational display. All details of the display are hard-coded within the implementation itself. There are no gauge-specific data tags.

Read Tags

Signature	Description	Type	Format

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
cnvd
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    pnvd
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- data tag ----
        pnvd
    <endo>
    <size> ---- x,y,xsize,ysize ----
    253
    495
    143
    116
    <lite> ---- light ----
    plef
    <help> ---- popup help ----
    Navigation Display
<endo> ==== END GAUGE ENTRY ====
```



comm Comm Radio Gauge

Class: CCommRadioGauge

Base Class: CBitmapGauge

Description:

Unknown. No examples in the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

comp Basic Compass Gauge

Class: CBasicCompassGauge

Base Class: CBitmapGauge

Description:

This common gauge type implements a traditional whiskey compass. The <mesg> tag should refer to a magnetic compass subsystem of type 'comp' from which the gauge will get the aircraft's magnetic heading.

The <comp> tag specifies the artwork for the compass; this is typically a horizontal strip which represents all 360 degrees, plus some overlap. Here is an example as used in the Flyhawk compass example below:



After retrieving the magnetic heading, the gauge calculates the x-pixel position of the current heading in the artwork strip, by using the <orgx> and <nton> tags which tell it the pixel location of "zero degrees", and the number of pixels in 360 degrees. The gauge then uses the window size as specified by the <wind> tag to extract a portion of the artwork bitmap, centered on the current heading which is then displayed at the window location.

Read Tags

Signature	Description	Type	Format
comp	Compass Strip Bitmap	ART	Compass strip (PBM)
wind	Window Size	INT	Pixel width of glass area
orgx	Origin 'X' for zero degrees	INT	Pixel location of first 0 deg mark on strip
nton	North to North pixel distance	INT	Number of pixels from 0 deg to 0 deg on strip

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
comp
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  comp
  <size> ---- size and location ----
  680
  22
  51
  16
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
    mcomp
  <endo>
  <comp> ---- compass strip bitmap ----
  compass.pbm
  <wind> ---- window width ----
```



```
51
<orgx> ---- origin of zero degrees offset ----
36
<nton> ---- north to north pixels ----
216
<over> ---- overlay ----
skyhcomp.pbm
0
0
<help> ---- popup help ----
Magnetic Compass
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhcpsm.raw
<endo> ==== END GAUGE ENTRY ====
```

comr Generic Comm Radio Gauge

Class: CGenericCommRadioGauge

Base Class: CBitmapGauge

Description:

This gauge type implements a generic Comm Radio. The <radi> tag specifies the ID and unit number of the associated Comm radio subsystem.

The active and standby frequencies are each displayed as clickable regions, with each digit being independently settable by clicking left/right on the digit. Click area objects for the active frequency are <ca01> and <ca02> for whole and fractional MHz respectively. Click areas for the standby frequency are <ca03> and <ca04>. The <ca05> (Comm Tune) and <ca06> Comm Transfer button click areas provide an alternative means of changing the frequency; right/left clicking on the Comm Tune click area changes the standby frequency, which can then be toggled with the active frequency by clicking on the Comm Transfer button.

If the <sdim> tag is included in the gauge specification, then the standby frequency is displayed in a dimmed font colour.

Read Tags

Signature	Description	Type	Format
radi	Radio Identifier	ID_TYPE	ADF radio subsystem ID
		INT	ADF radio subsystem unit number
ca01	Active Whole MHz	OBJECT	Click Area
ca02	Active Fractional MHz	OBJECT	Click Area
ca03	Standby Whole MHz	OBJECT	Click Area
ca04	Standby Fractional MHz	OBJECT	Click Area
ca05	Comm Tune	OBJECT	Click Area
ca06	Comm Transfer	OBJECT	Click Area
sdim	Standby Dimmed	NO_TYPE	

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
comr
<bgn> ==== BEGIN OBJECT ====
  <unid> ---- unique id ----
  com1
  <size> ---- x,y,xsize,ysize ----
  689
  617
  69
  71
  <radi> ---- radio identifier ----
  comm
  1
  <sdim> ---- standby digits dimmed ----
  <ca01> ---- COM active whole digits ----
  <bgn> ---- begin click area ----
```



```

        <rect> ---- click area rect ----
        15
        5
        33
        18
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        COM Active Frequency
    <endo> ---- end click area ----
    <ca02> ---- COM active fract digits ----
    <bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        35
        5
        49
        18
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        COM Active Frequency
    <endo> ---- end click area ----
    <ca03> ---- COM standby whole digits ----
    <bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        15
        18
        33
        30
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        COM Standby Frequency
    <endo> ---- end click area ----
    <ca04> ---- COM standby fract digits ----
    <bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        35
        18
        49
        30
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        COM Standby Frequency
    <endo> ---- end click area ----
    <ca05> ---- COM tune click area ----
    <bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        45
        52
        61
        68
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        COM Frequency Tune Knob
    <endo> ---- end click area ----
    <ca06> ---- COM transfer click area ----
    <bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        33
        46
        41
        56
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        COM Transfer Button
    <endo> ---- end click area ----
    <help> ---- popup help ----

```

COM1 Radio
<endo> ==== END OBJECT ====

cpfd Collins Primary Flight Display

Class: CCollinsPFDGauge

Base Class: CBitmapGauge

Description:

This gauge type implements the Collins Primary Flight Display, part of the Collins EFIS display package. The <mesg> tag must connect this gauge to a subsystem of type 'cpfd'.

There are no gauge-specific data tags associated with this gauge type; all elements of the display are hard-coded within the gauge object implementation.

Read Tags

Signature	Description	Type	Format

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
cpfd
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    ppfd
    <mesg>
    <bgno>
      <conn>
        ppfd
        <dtag>
        nsys
    <endo>
    <size> ---- x,y,xsize,ysize ----
    253
    332
    143
    116
    <lite> ---- light ----
    plef
    <help> ---- popup help ----
    Primary Flight Display
<endo> ==== END GAUGE ENTRY ====
```



cprk Cabin Pressure Rate Knob Gauge

Class: CCabinPressureRateKnobGauge

Base Class: CBitmapGauge

Description:

This gauge type implements a rate adjustment knob for cabin pressurization.

Read Tags

Signature	Description	Type	Format

Example (TRI34.PNL)

```
<gage> ===== GAUGE ENTRY =====
cprk
<bgno> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
    cprk
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
      pzat
    <user> ---- user data ----
      HARDWARE,OTHER
    <dtag> ---- user data ----
      rats
  <endo>
  <size> ---- x,y,xsize,ysize ----
    571
    371
    322
    183
  <bmap> ---- bitmap name ----
    kacpress.pbg
  <ofxy> ---- bitmap offset ----
    62
    0
  <alti> ---- altitude range ----
    -1000
    15000
  <rate> ---- rate range ----
    200
    2000
  <crsr> ---- cursor ----
    mpfinger.csr
  <knob> ---- rate knob ----
  <bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
      kacrate.pbg
    <rect> ---- click area rect ----
      0
      123
      62
      183
    <curs> ---- cursor ----
      mpfinger.csr
    <help> ---- popup help ----
      Cabin Pressure Rate
  <endo> ---- end click area ----
  <cabn> ---- altitude knob ----
```



```
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    kacpknob.pbg
    <rect> ---- click area rect ----
    256
    123
    322
    183
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Cabin Altitude
<endo> ---- end click area ----
<help> ---- popup help ----
Cabin Pressure Altitude
<endo> ==== END GAUGE ENTRY ====
```

cprs Cabin Pressure Gauge

Class: CCabinPressureGauge

Base Class: CBitmapGauge

Description:

This gauge type display cabin pressurization. The <mesg> tag must specify a cabin pressurization subsystem of type 'pzat'.

This gauge type is actually a type of triple-needle gauge, although it is not a descendent of the '3ned' gauge type. The actual cabin altitude is displayed using the needle specified by the <alti> tag; the pressure differential between cabin pressure and outside atmospheric pressure is specified by the <diff> needle tag, and the rate of change of cabin pressure is specified by the <rate> tag.

Read Tags

Signature	Description	Type	Format
alti	Cabin Altitude	OBJECT	Needle
diff	Pressure Differential	OBJECT	Needle
rate	Rate	OBJECT	Needle

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
cprs
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    cprs
  <size> ---- x,y,xsize,ysize ----
    1443
    519
    71
    71
  <user> ---- user data ----
    HARDWARE,GAUGE
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
      pzat
    <user> ---- user data ----
      HARDWARE,OTHER
    <dtag> ---- user data ----
      rats
  <endo>
  <alti> ---- first needle ----
  <bgno> ---- begin needle ----
    <cntr> ---- needle center x,y ----
      35
      35
    <sang> ---- starting angle ----
      0.0
    <gmin> ---- min value allowed ----
      0.0
    <gmax> ---- max value allowed ----
      50000.0
    <nedl> ---- bitmap needle ----
```



```

rhc3
<bgno> ==== BEGIN BITMAP NEEDLE ====
      <file> ---- needle file ----
      rhcpndl3.ndl
<endo> ==== END BITMAP NEEDLE ====
<gtbl> ---- gauge value table ----
<bgno> ---- begin table ----
      <fmt1> - x y
      0.0 0.0
      2000.0 26.4
      4000.0 52.6
      6000.0 77.3
      8000.0 103.1
      10000.0 128.9
      30000.0 153.8
      50000.0 180.0
      <endf> - end format
<endo> ---- end table ----
<endo> ---- end needle ----
<diff> ---- second needle ----
<bgno> ---- begin needle ----
      <cntr> ---- needle center x,y ----
      35
      35
      <sang> ---- starting angle ----
      0.0
      <gmin> ---- min value allowed ----
      0.0
      <gmax> ---- max value allowed ----
      10.0
      <nedl> ---- bitmap needle ----
      rhc1
      <bgno> ==== BEGIN BITMAP NEEDLE ====
            <file> ---- needle file ----
            rhcpndl1.ndl
      <endo> ==== END BITMAP NEEDLE ====
      <gtbl> ---- gauge value table ----
      <bgno> ---- begin table ----
            <fmt1> - x y
            0.0 0.0
            2.0 26.4
            4.0 52.6
            6.0 77.3
            8.0 103.1
            10.0 128.9
            <endf> - end format
      <endo> ---- end table ----
<endo> ---- end needle ----
<rate> ---- third needle ----
<bgno> ---- begin needle ----
      <cntr> ---- needle center x,y ----
      35
      35
      <sang> ---- starting angle ----
      208.4
      <gmin> ---- min value allowed ----
      -2000.0
      <gmax> ---- max value allowed ----
      2000.0
      <nedl> ---- bitmap needle ----
      rhc2
      <bgno> ==== BEGIN BITMAP NEEDLE ====
            <file> ---- needle file ----
            rhcpndl2.ndl
      <endo> ==== END BITMAP NEEDLE ====
      <gtbl> ---- gauge value table ----
      <bgno> ---- begin table ----
            <fmt1> - x y
            -2000.0 0.0
            -1000.0 18.7
            0.0 61.6

```

```
1000.0 104.6
2000.0 123.7
<endf> - end format
<endo> ---- end table ----
<endo> ---- end needle ----
<lite> ---- light ----
clfl
<mask> ---- light mask ----
rhmask20.raw
<help> ---- popup help ----
Cabin Pressure
<endo> ==== END GAUGE ENTRY ====
```

crtu **Collins Radio Tuning Unit**

Class: CCollinsRTU Gauge

Base Class: CBitmapGauge

Description:

This gauge type implements the Collins Radio Tuning Unit display, part of the Collins EFIS display suite. The <mesg> tag must connect this gauge to a subsystem of type 'crtu'.

Many click area data tags are defined which enable the user to tune various radio frequencies by clicking directly on the displayed values.

Read Tags

Signature	Description	Type	Format
lin1	Line 1	OBJECT	Rectangle
lin2	Line 2	OBJECT	Rectangle
lin3	Line 3	OBJECT	Rectangle
lin4	Line 4	OBJECT	Rectangle
lin5	Line 5	OBJECT	Rectangle
btn1	Button 1	OBJECT	Click Area
btn2	Button 2	OBJECT	Click Area
btn3	Button 3	OBJECT	Click Area
btn4	Button 4	OBJECT	Click Area
btn5	Button 5	OBJECT	Click Area
cmaw	COM Active Whole MHz	OBJECT	Click Area
cmaf	COM Active Fract MHz	OBJECT	Click Area
cmsw	COM Stby Whole MHz	OBJECT	Click Area
cmsf	COM Stby Fract MHz	OBJECT	Click Area
nvaw	NAV Active Whole MHz	OBJECT	Click Area
nvaf	NAV Active Fract MHz	OBJECT	Click Area
adf1	ADF Digit 1	OBJECT	Click Area
adf2	ADF Digit 2	OBJECT	Click Area
adf3	ADF Digit 3	OBJECT	Click Area
adf4	ADF Digit 4	OBJECT	Click Area
adf5	ADF Digit 5	OBJECT	Click Area
xpd1	Transponder Digit 1	OBJECT	Click Area
xpd2	Transponder Digit 2	OBJECT	Click Area
xpd3	Transponder Digit 3	OBJECT	Click Area
xpd4	Transponder Digit 4	OBJECT	Click Area
rtu_	RTU Subsystem	OBJECT	Message
dme1	DME 1 Subsystem	OBJECT	Message
dme2	DME 2 Subsystem	OBJECT	Message
dmeH	DME Hold Announcer	ART	PBM filename
		INT	X pixel size
		INT	Y pixel size
repl	XPDR Reply Announcer	ART	PBM filename
		INT	X pixel size
		INT	Y pixel size
atci	ATC ID	OBJECT	Click Area
dmeh	DME Hold Button	OBJECT	Click Area
coms	COM Squelch	OBJECT	Click Area
atcs	ATC Standby	OBJECT	Click Area
1/2_	1 / 2	OBJECT	Click Area

adft	ADF Test	OBJECT	Click Area
side	Side	INT	

Example (TRI40.PNL)

Due to the extreme length and repetitive nature of this gauge type, an example has not been included. There are examples of this gauge type in file TRI40.PNL within TRI4.POD.



cswi Covered Switch

Class: CCoveredSwitch

Base Class: CBitmapGauge

Description:

This interactive gauge type is intended to implement a switch which has a safety cover that must be flipped up before the switch can be toggled.

Read Tags

Signature	Description	Type	Format

Example (TRI226.PNL)

```
<gage> ==== GAUGE ENTRY ====
cswi
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    aviE
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- data tag ----
        aviE
        <user> ---- user data ----
        HARDWARE, SWITCH
    <endo>
    <size> ---- x,y,xsize,ysize ----
    300
    522
    80
    159
    <bmap> ---- bitmap name ----
    navjeacb.pbg
    <csru> ---- up cursor ----
    mpflipup.csr
    <csrd> ---- down cursor ----
    mpflipdn.csr
    <sfxu> ---- up sfx ----
    switchup.wav
    <sfxd> ---- down sfx ----
    switchdn.wav
    <sstr> ---- switch strings ----
    2
    Off
    On
    <help> ---- popup help ----
    Emergency Avionics Bus
<endo> ==== END GAUGE ENTRY ====
```



cvsp Collins V-Speed Panel

Class: CCollinsVSpeedPanel

Base Class: CBitmapGauge

Description:

This panel is part of the Collins EFIS avionics suite equipped in the TRI Peregrine 800TR jet. This panel allows the setting of the V-speeds displayed on the EFIS Primary Flight Display.

The <vsel>

Read Tags

Signature	Description	Type	Format
sped	V-Speed Setting Knob	OBJECT	Click Area
vsel	V-Speed Advance	OBJECT	Click Area

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
cvsp
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    pvsp
    <size> ---- x,y,xsize,ysize ----
    186
    333
    31
    56
    <mesg> ---- switch message ----
    <bgno> ---- begin ----
      <conn> ---- group ----
      ppfd
      <user> ---- user data ----
      HARDWARE,OTHER
      <user> ---- user data ----
      UNIT,1
      <dtag> ---- user data ----
      nsys
    <endo> ---- end ----
    <sped> ---- speed setting knob ----
    <bgno> ---- begin ----
      <rect> ---- rect ----
      5
      33
      24
      52
      <curs> ---- cursor ----
      mpfinger.csr
      <help> ---- help ----
      V-Speed Setting
    <endo> ---- end ----
    <vsel> ---- V select ----
    <bgno> ---- begin ----
      <rect> ---- rect ----
      7
      6
      21
      19
      <curs> ---- cursor ----
      mpfinger.csr
```



```
          <help> ---- help ----  
          V-Speed Advance  
<endo> ---- end ----  
<help> ---- help ----  
Pilot V-Speed Panel  
<endo> ==== END GAUGE ENTRY ====
```

dgrg Digital Readout Gauge

Class: CDigitalReadoutGauge

Base Class: CBitmapGauge

Description:

This common gauge type implements a generic digital readout.

The artwork for the digits is specified with the <wdig> and <fdig> tags respectively, for whole and fractional parts of the readout value. This allows, for example, the fractional part to be drawn smaller than the whole part, by creating the appropriate artwork. Each digit of the fractional artwork should also include the decimal point (this means that the gauge can only effectively display values to a precision of one decimal point).

The placement of the digits is determined by the <digY> and <digX> tags. You should ensure that you define enough digits in the <digX> tag to accomodate the maximum length of the value that can be displayed by the gauge.

The <dfmt> tag (which is inherited from an ancestor gauge type) determines how the data value retrieved from the <mesg> subsystem is actually formatted for display.

Read Tags

Signature	Description	Type	Format
wdig	Whole digits	ART	PBG filename
fdig	Fractional digits	ART	PBG filename
digY	Digit Y offset	INT	Y pixel offset
digX	Digit X offsets	INT	Number of digits
		INT	X offset digit 1
		INT	X offset digit 2
		...	... for number of digits

Example (MALI00.PNL)

```
<gage> ==== GAUGE ENTRY ====
dgrg
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  volt
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data connection ----
    volt
    <user> ---- user data ----
    HARDWARE, GAUGE
  <endo>
  <size> ---- placement and size ----
  863
  500
  84
  29
  <wdig> ---- whole digits ----
  maliwfnt.pbg
  <fdig> ---- fract digits ----
```



```
malidfnt.pbg
<digY> ---- digit Y offset ----
7
<digX> ---- digit X offset ----
4
63
45
27
8
<dfmt> ---- display format ----
%.1f volts
<help> ---- popup help ----
System Voltage
<endo> ==== END GAUGE ENTRY ====
```

dme2 DME Gauge

Class: CDME2IndicatorGauge

Base Class: CGauge

Description:

This gauge type implements a Distance Measuring Equipment display. Its operation is similar to the other DME gauge which uses the type 'dmei'.

Read Tags

Signature	Description	Type	Format
hold	Hold Mode	NO_TYPE	
left	DME Text	INT	X pixel offset
		INT	Y pixel offset
right	SEL Text	INT	X pixel offset
		INT	Y pixel offset
ch__	Channel Button	OBJECT	Click Area
sel__	Select Button	OBJECT	Click Area
pwr__	Power Button	OBJECT	Click Area

Example (TRI30.PNL)

```
<gage> ==== DME Indicator ====
dme2
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    dme1
  <size> ---- x,y,xsize,ysize ----
    424
    745
    86
    42
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data provider ----
      dme1
    <user> ---- user data ----
      HARDWARE,OTHER
    <dtag> ---- user data ----
      mode
  <endo>
  <hold> ---- has Hold mode ----
  <left> ---- DME text x,y ----
    18
    8
  <right> ---- SEL text x,y ----
    48
    8
  <ch__> ---- CH button ----
  <bgno> ---- begin ----
    <bmap> ---- bitmap ----
      kadmech.pbg
    <rect> ---- click rect ----
      12
      29
      28
      37
    <curs> ---- cursor ----
      mpfinger.csr
  <help> ---- help ----
```



```

        Channel
<endo> ---- end ----
<sel_> ---- SEL button ----
<bgn> ---- begin ----
        <bmap> ---- bitmap ----
        kadmes1.pbg
        <rect> ---- click rect ----
        35
        29
        50
        37
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Select
<endo> ---- end ----
<pwr_> ---- PWR button ----
<bgn> ---- begin ----
        <bmap> ---- bitmap ----
        kadmepw.pbg
        <rect> ---- click rect ----
        56
        29
        71
        37
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Power
<endo> ---- end ----
<help> ---- popup help ----
        DME Indicator
<endo> ===== END GAUGE ENTRY =====

```

dmei DME Gauge

Class: CDMEIndicatorGauge

Base Class: CGauge

Description:

This is an alternative Distance Measuring Indicator gauge type from the other type 'dme2'.

Read Tags

Signature	Description	Type	Format
hold	Hold Mode	NO_TYPE	
dist	Distance Text	INT	X pixel offset
		INT	Y pixel offset
sped	Speed Text	INT	X pixel offset
		INT	Y pixel offset
time	Time Text	INT	X pixel offset
		INT	Y pixel offset
knob	Mode Knob	OBJECT	Knob
tiny	Use Tiny fonts	NO_TYPE	

Example (TRI200.PNL)

```
<gage> ==== DME Indicator ====
dmei
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    dmei
  <size> ---- x,y,xsize,ysize ----
    27
    222
    67
    33
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data provider ----
      dmep
    <user> ---- user data ----
      HARDWARE,OTHER
    <dtag> ---- user data ----
      mode
  <endo>
  <hold> ---- has Hold mode ----
  <dist> ---- distance text x,y ----
    0
    3
  <sped> ---- speed text x,y ----
    23
    3
  <time> ---- time text x,y ----
    45
    3
  <knob> ---- Hold knob ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
      21
      15
      43
      30
    <bmap> ---- bitmap ----
```



```
        pndme.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        DME Mode Selector
<endo> ---- end click area ----
<tiny> ---- use tiny fonts ----
<lite> ---- light ----
alit
<help> ---- popup help ----
DME Indicator
<endo> ==== END GAUGE ENTRY ====
```

dned **Digital Needle Gauge**

Class: CDigitalNeedleGauge

Base Class: CNeedleGauge

Description:

This gauge has been used extensively in the Pilatus and the Bell 407. It implements a powered digital display that would typically indicate the needle position by illuminating LED/LCD segments around the circumference of the gauge (rather than by the traditional pointer needle).

Read Tags

Signature	Description	Type	Format
nndl	Night Needle Bitmap	OBJECT	Needle
nbak	Night backdrop	ART	Backdrop file (PBM)
		INT	X position
		INT	Y position
powr	Power to gauge	OBJECT	Message

Example (PIL_MAIN.PNL)

Note: The 'dned' gauge type displays only the outer digital needle portion of the gauge; the inner numeric gauge value is displayed with the 'drog' (Digital Readout) gauge type.

```
<gage> ==== GAUGE ENTRY ====
dned
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  torq
  <size> ---- x,y,xsize,ysize ----
  580
  268
  74
  74
  <cntr> ---- needle center x,y ----
  37
  37
  <mesg> -- Subsystem Message --
  <bgno>
    <conn>
    trqi
    <dtag>
    indn
  <endo>
  <nndl> ---- bitmap needle ----
  digi
  <bgno> ==== BEGIN BITMAP NEEDLE ====
    <file> ---- needle ----
    pdigindl.ndl
  <endo> ==== END BITMAP NEEDLE ====
  <gmax> ---- max value allowed ----
  61.0
  <gtbl> ---- gauge value table ----
  <bgno> ---- begin table ----
    <fmt1> - x y
    0.0 3.2
    10.0 51.3
```



```
20.0 100.1
30.0 149.6
40.0 199.4
50.0 251.0
60.0 298.1
70.0 343.1
<endf> - end format
<endo> ---- end table ----
<help> ---- help string ----
Engine Torque (psi)
<lite> ---- light flag ----
silt
<dfmt> ---- display format ----
%.1f psi
<endo> ==== END GAUGE ENTRY ====
```

drog Digital Readout Gauge

Class: CDigitsReadoutGauge

Base Class: CGauge

Description:

This gauge type is similar to the Digital Readout Gauge (type 'dgrg'). This gauge type uses a standard LED-like font to display the gauge value. The font colour can be specified using the <frgb> tag, and the justification of the readout text within the gauge boundaries is controlled via the <just> tag.

Read Tags

Signature	Description	Type	Format
frgb	Font RGB	INT	Red value
		INT	Green value
		INT	Blue value
just	Text Justification	WORD	RIGHT LEFT CENTER

Example (PIL_MAIN.PNL)

Note: The 'drog' gauge type displays on the numeric gauge value in the center of the gauge; the outer digital needle portion is displayed with the 'dned' (Digital Needle) gauge type.

```
<gage> ==== GAUGE ENTRY ====
drog
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    dNg_
  <size> ---- x,y,xsize,ysize ----
    727
    300
    20
    11
  <frgb>
    255
    255
    255
  <just> -- Justify text --
    RIGHT
  <mesg> -- Subsystem Message --
  <bgno>
    <conn>
      _ngd
    <dtag>
      indn
  <endo>
  <gmin> ---- min value allowed ----
    0.0
  <gmax> ---- max value allowed ----
    104.0
  <help> ---- help string ----
    Generator Tachometer
  <lite> ---- light flag ----
    silt
  <dfmt> ---- display format ----
```



%.1f
<endo> ==== END GAUGE ENTRY =====

dswi Dual Switch

Class: CDualSwitchGauge

Base Class: CGauge

Description:

This gauge type allows two independent switches to be integrated into one gauge object.

Read Tags

Signature	Description	Type	Format
nums	Number of Switch Position	INT	
orie	Orientation	INT	1 = Vertical 2 = Horizontal
usrl	Left Message	OBJECT	Message
usrr	Right Message	OBJECT	Message
mmnt	Momentary Contact	NO_VALUE	

Example (TRI210.PNL)

```
<gage> ===== GAUGE ENTRY =====
dswi
<bgno> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
  strt
  <nums> ---- num switch states ----
  3
  <mmnt> ---- momentary switch ----
  <orie> ---- orientation ----
  2
  <usrl> ---- left user data ----
  <bgno> ---- begin ----
    <grou> ---- group ----
    strt
    <user> ---- user data ----
    HARDWARE, SWITCH
    <user> ---- user data ----
    ENGINE, 1
  <endo> ---- end ----
  <usrr> ---- right user data ----
  <bgno> ---- begin ----
    <grou> ---- group ----
    strt
    <user> ---- user data ----
    HARDWARE, SWITCH
    <user> ---- user data ----
    ENGINE, 2
  <endo> ---- end ----
  <size> ---- x,y,xsize,ysize ----
  296
  341
  44
  30
  <bmap> ---- bitmap name ----
  pnsWSTRt.pbg
  <csrl> ---- left cursor ----
  mpfliPlt.csr
  <csrr> ---- right cursor ----
  mpfliPrT.csr
  <sfxl> ---- left sfx ----
```



```
switchup.wav
<sfx> ---- right sfx ----
switchdn.wav
<sstr> ---- switch strings ----
3
Off
Neutral
Start
<help> ---- popup help ----
Engine Starter
<endo> ==== END GAUGE ENTRY ====
```

dtch Dual Tachometer Needle Gauge

Class: CDualTachometerGauge

Base Class: CTwoNeedleGauge

Description:

The Dual Tachometer gauge type is a specialized dual needle gauge intended for displaying the engine RPM for a twin-engine aircraft. It does not appear to have any gauge-specific tags.

Read Tags

Signature	Description	Type	Format

Example (TRI200.PNL)

```
<gage> ==== GAUGE ENTRY ====
dtch
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    rpms
  <size> ---- placement and size ----
    622
    65
    95
    95
  <ndl1> ---- engine 1 rpm needle ----
  <bgn> ---- BEGIN NEEDLE ----
    <mesg>
    <bgn>
      <conn>
      rpm1
    <dtag>
      rpm1
    <endo>
  <cntr> ---- needle center x,y ----
    46
    47
  <ned1> ---- bitmap needle ----
  std
  <bgn> ==== BEGIN BITMAP NEEDLE ====
    <shar> ---- shared needle ----
    <offx> ---- offset X ----
      35
    <offy> ---- offset Y ----
      35
    <base> ---- base bitmap name ----
      NEEDLE1
  <endo> ==== END BITMAP NEEDLE ====
  <gtbl> ---- gauge value table ----
  <bgn> ---- begin table ----
    <fmt1> - x y
      0.0 0.0
      500.0 38.0
      1000.0 75.0
      1500.0 112.5
      2000.0 149.5
      2500.0 185.5
      3000.0 222.5
      3500.0 260.0
      3600.0 275.0
    <endf> - end format
```



```

<endo> ---- end table ----
<sang> ---- starting angle ----
228.0
<gmin> ---- min value allowed ----
0.0
<gmax> ---- max value allowed ----
3600.0
<dfmt> ---- display format ----
%.1f RPM (Left)
<endo> ---- END NEEDLE ----
<nd12> ---- engine 2 rpm needle ----
<bgn0> ---- BEGIN NEEDLE ----
<mesg>
<bgn0>
    <conn>
    rpm2
    <dtag>
    rpm2
<endo>
<cntr> ---- needle center x,y ----
46
47
<ned1> ---- bitmap needle ----
stnd
<bgn0> ==== BEGIN BITMAP NEEDLE ====
    <shar> ---- shared needle ----
    <offx> ---- offset X ----
    35
    <offy> ---- offset Y ----
    35
    <base> ---- base bitmap name ----
    NEEDLE1
<endo> ==== END BITMAP NEEDLE ====
<gtbl> ---- gauge value table ----
<bgn0> ---- begin table ----
    <fmt1> - x y
    0.0 0.0
    500.0 38.0
    1000.0 75.0
    1500.0 112.5
    2000.0 149.5
    2500.0 185.5
    3000.0 222.5
    3500.0 260.0
    3600.0 275.0
    <endf> - end format
<endo> ---- end table ----
<sang> ---- starting angle ----
228.0
<gmin> ---- min value allowed ----
0.0
<gmax> ---- max value allowed ----
3600.0
<dfmt> ---- display format ----
%.1f RPM (Right)
<endo> ---- END NEEDLE ----
<lite> ---- light ----
plit
<mask> ---- mask ----
pnmask16.raw
<help> ---- popup help ----
Tachometer
<endo> ==== END GAUGE ENTRY ====

```

dual Dual Knob

Class: CDualKnob

Base Class: CGauge

Description:

This gauge type implements a dual knob device, where there is typically an inner and outer knob which each independently control two different settings. There are no examples of this gauge type in any of the default aircraft, however it is the parent type of gauge type 'shlk' (Skyhawk Light Knob) which illustrates its use.

Read Tags

Signature	Description	Type	Format

Example

egtg Exhaust Gas Temperature Needle Gauge

Class: CExhaustGasTempGauge

Base Class: CNeedleGauge

Description:

This specialized needle gauge is intended to display the Exhaust Gas Temperature (EGT) of a piston engine. In addition to all of the generic needle gauge functionality which is inherited from its parent type 'need', the EGT gauge also features a reference needle which can be set by the user.

The artwork for the indicator needle uses the normal <nedl> tag, and for the reference needle as specified in the <egtn> tag. The <egtr> tag supplies a default value for the position of the reference needle when the sim starts up.

Read Tags

Signature	Description	Type	Format
egtr	Reference Needle Default	REAL	Default EGT setting for reference needle
egtn	Reference Needle	ART	PBG filename

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
egtg
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    eEGT
  <size> ---- x,y,xsize,ysize ----
    109
    285
    33
    66
  <cntr> ---- needle center x,y ----
    13
    32
  <over> ---- overlay image ----
    skyhegt.pbm
    0
    0
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- connected to ----
    eEGT
    <user> ---- user data ----
    HARDWARE, GAUGE
    <user> ---- user data ----
    ENGINE,1
  <endo>
  <nedl> ---- bitmap needle ----
  smal
  <bgn> ==== BEGIN BITMAP NEEDLE ====
    <file> ---- needle file ----
    needle3.ndl
  <endo> ==== END BITMAP NEEDLE ====
  <gtbl> ---- gauge value table ----
  <bgn> ---- begin table ----
```



```
        <fmt1> - x y
        0.0 161.7
        100.0 148.8
        200.0 116.2
        300.0 78.7
        400.0 37.9
        500.0 22.5
        <endf> - end format
<endo> ---- end table ----
<gmin> ---- min value allowed ----
0.0
<gmax> ---- max value allowed ----
500.0
<egtr> ---- EGT reference needle default ----
200.0
<egtn> ---- EGT reference needle bitmap ----
needle6.pbg
<help> ---- popup help ----
Exhaust Gas Temperature
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhegtm.raw
<curs> ---- cursor ----
mpfinger.csr
<endo> ==== END GAUGE ENTRY ====
```

elt_ Two-State ELT Switch

Class: CTwoStateELTGauge

Base Class: CSimpleSwitchGauge

Description:

This gauge type implements one type of Emergency Location Transmitter switch (the other is gauge type 'selt'). This is a simple two-state On/Off switch and does not have any gauge-specific data tags. See the generic switch gauge type 'swit' for details of the inherited tags.

Read Tags

Signature	Description	Type	Format

Example (MALI00.PNL)

```
<gage> ==== GAUGE ENTRY ====
elt_
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    elts
  <size> ---- x,y,xsize,ysize ----
    1586
    401
    21
    36
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn>
    elts
    <dtag>
    st8t
  <endo>
  <bmap> ---- bitmap name ----
  pmeltsw.pbg
  <csru> ---- up cursor ----
  mpflipup.csr
  <csrd> ---- down cursor ----
  mpflipdn.csr
  <sfxu> ---- up sfx ----
  switchup.wav
  <sfxd> ---- down sfx ----
  switchdn.wav
  <help> ---- popup help ----
  Emergency Locator
<endo> ==== END GAUGE ENTRY ====
```



fdhz Artificial Horizon / Flight Director Gauge

Class: CFDHorizonGauge Gauge

Base Class: CBitmapGauge

Description:

This gauge type implements an artificial horizon gauge that includes an autopilot Flight Director indicator. Most of the data tags are common with the other artificial horizon gauge type 'hori' (which does not have the Flight Director bars). The artwork for the FD overlay is specified by the <bars> tag.

Read Tags

Signature	Description	Type	Format
foot	Football artwork (football-shaped horizon bar)	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
horz	Horizon artwork (Horizon needle or small airplane symbol)	ART	Horizon file (PBG)
		INT	X pixel offset
		INT	Y pixel offset
bars	Flight Director Bars artwork	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
hoff	Horizon offset (Default horizon needle offset in Y axis)	INT	Pixels (default 0)
irim	Inner rim artwork (Ring with bank markings)	ART	Inner rim file (PBG)
		INT	X coordinate
		INT	Y coordinate
orim	Outer rim artwork (Sometimes a separate piece of art is needed for an outer ring)	ART	Outer rim file (PBM)
		INT	X coordinate
		INT	Y coordinate
fron	Front plate artwork (Front face plate for the instrument)	ART	Front plate file (PBM)
		INT	X coordinate
		INT	Y coordinate
pixd	Pixels per degree (football) (Use this to calibrate the pitch reading)	REAL	

moff	Maximum drawing offset	INT	Pixels
knob	Horizon needle adjustment knob	OBJECT	Knob
flag	Gyro flag name?	ART	Flag file (PBM)
pich	Pitch message	OBJECT	Message
roll	Roll message	OBJECT	Message

Example (TRI200.PNL)

```

<gage> ===== GAUGE ENTRY =====
fdhz
<bgno> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
    hori
  <size> ---- x,y,xsize,ysize ----
    234
    161
    90
    90
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- autopilot ----
    AXIS
    <user> ---- user data ----
    HARDWARE, OTHER
    <dtag> ---- user data ----
    _FCS
  <endo>
  <pich> -- Pitch Message --
  <bgno>
    <conn>
    att1
    <dtag>
    pich
  <endo>
  <roll> -- Roll Message --
  <bgno>
    <conn>
    att1
    <dtag>
    roll
  <endo>
  <foot> ---- football filename ----
  ahoriz_b.pbg
  2
  2
  <horz> ---- horizon needle filename ----
  fdhoriz.pbm
  -1
  4
  <irim> ---- inner rim filename ----
  ahoriz_d.pbg
  3
  2
  <orim> ---- outer rim filename ----
  ahoriz_f.pbm
  2
  3
  <bars> ---- FD bars ----
  ahoriz_g.pbg
  2
  2
  <over> ---- overlay image ----
  navjatto.pbm
  -9
  -10
  <pixd> ---- pixels per degree ----
  0.6

```



```

<hoff> ---- horizon offset ----
-1
<moff> ---- maximum pixel offset ----
23
<help> ---- popup help ----
Attitude Indicator
<knob> ---- horizon cross knob ----
<bgn0> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhknob.pbg
    <rect> ---- click area rect ----
    36
    71
    54
    89
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Horizon Needle Adjustment
<endo> ---- end click area ----
<lite> ---- light ----
plit
<mask> ---- mask ----
pnmask06.raw
<endo> ==== END GAUGE ENTRY ====

```

fFlw **Fuel Flow Needle Gauge**

Class: CFuelFlowGauge

Base Class: CNeedleGauge

Description:

This needle gauge is intended to display engine fuel flow. It does not appear to have any gauge-specific data tags.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ===== GAUGE ENTRY =====
fFlw
<bgno> ===== BEGIN GAUGE ENTRY =====
    <unid> ---- unique id ----
    flow
    <size> ---- x,y,xsize,ysize ----
    142
    285
    33
    66
    <cntr> ---- needle center x,y ----
    18
    32
    <over> ---- overlay image ----
    skyhfflw.pbm
    0
    0
    <mesg> -- connect to Fuelflow Subsystem --
    <bgno>
        <conn>
        fFlw
        <dtag>
        gph_
    <endo>
    <nedl> ---- bitmap needle ----
    smal
    <bgno> ===== BEGIN BITMAP NEEDLE =====
        <file> ---- needle file ----
        needle3.ndl
    <endo> ===== END BITMAP NEEDLE =====
    <gtbl> ---- gauge value table ----
    <bgno> ---- begin table ----
        <fmt1> - x y
        0.0 197.6
        5.0 203.3
        10.0 242.9
        15.0 290.2
        19.0 339.7
        <endf> - end format
    <endo> ---- end table ----
    <gmin> ---- min value allowed ----
    0.0
    <gmax> ---- max value allowed ----
    19.0
    <help> ---- popup help ----
    Fuel Flow
    <lite> ---- light flag ----
```



```
gage
<mask> ---- light mask ----
skyhfflm.raw
<endo> ==== END GAUGE ENTRY ====
```

find Flaps Indicator Gauge

Class: CFlapsIndicatorGauge

Base Class: CIndicatorGauge

Description:

This is a specialized type of indicator gauge (see gauge type 'indi') for use as a flaps position indicator. It does not appear to have any gauge-specific data tags.

Read Tags

Signature	Description	Type	Format

Example (TRI41.PNL)

```
<gage> ==== GAUGE ENTRY ====
find
<bgno> ==== BEGIN OBJECT ====
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- data tag ----
        flap
        <user> ---- user data ----
        HARDWARE, SWITCH
        <dtag> ---- user data ----
        angl
    <endo>
    <size> ---- size and pos ----
    607
    236
    13
    51
    <gtbl> ---- gauge value table ----
    <bgno> ---- begin table ----
        <fmt1> - x y
        0.0 0.0
        15.0 5.0
        25.0 8.0
        40.0 14.0
        <endf> - end format
    <endo> ---- end table ----
    <bmap> ---- bitmaps ----
    rhflpind.pbg
    <help> ---- popup help ----
    Flaps Indicator
<endo> ==== END OBJECT ====
```



flap Flaps Switch

Class: CFlapsSwitchGauge

Base Class: CPushPullKnobGauge

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
flap
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  flap
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- data tag ----
    flap
    <user> ---- user data ----
    HARDWARE, SWITCH
  <endo>
  <size> ---- x,y,xsize,ysize ----
  903
  545
  80
  110
  <bmap> ---- bitmap name ----
  skyhflap.pbg
  <crsr> ---- default cursor ----
  mpslide.csr
  <valu> --- click value range ----
  4
  0
  28
  0.0
  NONE
  29
  44
  1.0
  NONE
  45
  59
  2.0
  NONE
  60
  88
  3.0
  NONE
  <help> ---- popup help ----
  Flaps Switch
<endo> ==== END GAUGE ENTRY ====
```



fswi Simple Float Switch

Class: CSimpleFloatSwitchGauge

Base Class: CGauge

Description:

This switch type has discrete states, but each state causes a specific floating-point value to be sent to the connected subsystem. This is different from the behaviour of the generic 'swit' switch which sends integer values to its connected subsystem.

Read Tags

Signature	Description	Type	Format
bmap	Switch bitmap	ART	PBG filename
csru	Cursor - Switch Up	ART	CSR filename
csrd	Cursor - Switch Down	ART	CSR filename
sfxu	Sound Effect - Switch Up	WAV	WAV filename
sfxd	Sound Effect - Switch Down	WAV	WAV filename
stat	Switch States	INT	Number of states
		REAL	First state value
		REAL	Second state value
		...	...for number of states
sstr	State Strings	INT	Number of states
		WORD	Help for first state
		WORD	Help for second state
		...	...for number of states

Example (TRI200.PNL)

```
<gage> ==== GAUGE ENTRY ====
fswi
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  hilo
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
    hilo
    <user> ---- hardware ----
    HARDWARE,STATE
    <dtag> ---- data tag ----
    lev1
  <endo>
  <size> ---- x,y,xsize,ysize ----
  937
  31
  21
  30
  <bmap> ---- bitmap name ----
  pnanhilo.pbg
  <csru> ---- up cursor ----
  mpflipup.csr
  <csrd> ---- down cursor ----
  mpflipdn.csr
  <sfxu> ---- up sfx ----
```



```
switchup.wav
<sfxd> ---- down sfx ----
switchdn.wav
<stat> ---- states ----
2
0.7
1.0
<sstr> ---- state text ----
2
Low
High
<help> ---- popup help ----
Annunciator Hi/Lo
<endo> ==== END GAUGE ENTRY ====
```

fuel Fuel Quantity Needle Gauge

Class: CFuelGauge

Base Class: CNeedleGauge

Description:

This specialized needle gauge is intended to display fuel quantity values. It does not appear to have any gauge-specific data tags.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ===== GAUGE ENTRY =====
fuel
<bgno> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
    lfue
  <over> ---- overlay image ----
    skyhftl.pbm
    0
    0
  <size> ---- placement and size ----
    30
    285
    33
    66
  <mesg>
  <bgno>
    <conn>
    LqTy
    <dtag>
    gals
  <endo>
  <nedl> ---- bitmap needle ----
    smal
  <bgno> ===== BEGIN BITMAP NEEDLE =====
    <file> ---- needle file ----
    needle3.ndl
  <endo> ===== END BITMAP NEEDLE =====
  <cntr> ---- needle center x,y ----
    13
    33
  <gtbl> ---- gauge value table ----
  <bgno> ---- begin table ----
    <fmt1> - x y
    0.0 154.0
    5.0 128.7
    10.0 102.2
    15.0 75.6
    20.0 51.2
    26.0 26.2
    <endf> - end format
  <endo> ---- end table ----
  <gmin> ---- min value allowed ----
    0.0
  <gmax> ---- max value allowed ----
    26.0
  <help> ---- popup help ----
  Fuel Quantity (Left)
```



```
<lite> ---- light flag ----  
gage  
<mask> ---- light mask ----  
skyhflm.raw  
<endo> ==== END GAUGE ENTRY ====
```

gage Generic Gauge

Class: CGauge

Base Class: None

Description:

The 'gage' type serves as the parent class for all other gauge types. Note that the Base Class is "None" which is unique for all gauges. The data tags defined in this type are inherited by all other gauge types.

A unique identifier for all gauges must be provided via the <unid> data tag. ID values are typical Fly! 4-character identifiers such as 'alt1', 'nav1' etc. The identifier must be unique only within the panel that it is defined in; it is OK for a gauge on a different panel (i.e. different .PNL, .VFR, .VFN, .IFR, .IFN file) to use the same unique ID.

The <size> tag specifies the top-left and bottom-right position of the gauge relative to the top-left of the panel backdrop image.

The <cntr> tag is not used very often; it specifies the center of the gauge if it is different from the actual center pixel position. ??Unclear why the center of the gauge must be known...for needle rotations??

The <mesg> tag is a very important data tag which provides the linkage between the gauge which displays a data value, and an underlying subsystem that provides the data value being displayed. For example, an RPM gauge doesn't inherently know anything about the engine RPM; instead the <mesg> tag is used to "connect" the gauge to the appropriate subsystem type which is defined in the aircraft's .AMP (or other) file. The <mesg> tag is the preferred way to connect a gauge to its data provider; Fly! supported some additional tags which should not be used (though you will find examples of them in some standard Fly! 2 aircraft).

The <gmin> and <gmax> tags provide minimum and maximum values that can be displayed by the gauge. The value returned by the data provider is clamped to either of these two limiting values before being displayed by the gauge.

The <gtbl> allows an arbitrary mapping function to be performed by the gauge. For example, a temperature display gauge which displayed temperature in degrees Celsius, but was connected to a data provider which supplied the temperature in degrees Fahrenheit, could implement the F-to-C conversion via the <gtbl> data tag.

Read Tags

Signature	Description	Type	Format
unid	Unique ID	ID_TYPE	
size	Gauge size	INT	Top-left X position
		INT	Top-left Y position

		INT	Bottom-right X position
		INT	Bottom-right Y position
cntr	Center of gauge	INT	X center
		INT	Y center
mesg	Message data	OBJECT	Message
gmin	Gauge minimum value	REAL	
gmax	Gauge maximum value	REAL	
gtbl	Gauge table	OBJECT	Table
lite	Gauge light	ID_TYPE	Light ID
mask	Light mask	ART	Mask file (RAW)
fore	Gauge drawn in foreground or background?	INT	0 = background 1 = foreground
curs	Custom mouse cursor	FILE	Cursor file (CSR)
help	Popup help message	SENTENCE	
dtyp	Data type	INT	0 = CHAR 3 = INT 5 = REAL
dfmt	Display format	SENTENCE	C-style format spec (e.g. %.1f RPM)
nomc	No mouse clicks	NO_VALUE	
igno	Ignore match (Do not do a complete message verification on data changed?)	NO_VALUE	

Example

Examples of the data tags for the 'gage' object can be seen in all other gauge types since they are all descendants of this object type. This gauge type is never actually instantiated; in C++ terms it could be considered an abstract class type.

ghst Ghost Gauge

Class: CGhostGauge

Base Class: CNeedleGauge

Description:

Unknown. No examples of this gauge type exist in the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

gslv Gyro Slaving Switch

Class: CGyroSlavingGauge

Base Class: CBitmapGauge

Description:

This switch type is used to implement a gyro slaving function, which both displays the current gyro state and allows the user to change the setting.

Read Tags

Signature	Description	Type	Format
err_	Error Window	OBJECT	Rectangle
mode	Slave/Free Toggle Button	OBJECT	Click Area
ccw_	Counter-Clockwise Switch	OBJECT	Click Area
cw_	Clockwise Switch	OBJECT	Click Area
lrgb	Line RGB Colour	INT	Red colour
		INT	Green colour
		INT	Blue colour
sfx_	Sound Effects	WAV	WAV filename (Up)
		WAV	WAV filename (Down)

Example (MALI00.PNL)

```
<gage> ==== GAUGE ENTRY ====
gslv
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    gslv
    <mesg>
    <bgno>
        <conn>
        hsi1
        <dtag>
        yaw_
    <endo>
    <size> ---- x,y,xsize,ysize ----
    29
    164
    43
    52
    <err_> ---- error window ----
    <bgno> ---- begin button ----
        <rect> ---- rectangle ----
        14
        5
        24
        11
    <endo> ---- end button
    <slvb> ---- slave switch bitmap ----
    pmgyslav.pbg
    1
    16
    <mode> ---- Slave/Free toggle ----
    <bgno> ---- begin button ----
        <rect> ---- rectangle ----
        0
        14
```



```

43
31
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Slave/Free
<endo> ---- end button
<ccwb> ---- adjust switch bitmap ----
pmgyccw.pbg
2
34
<ccw_> ---- CounterClockwise Switch ----
<bgno> ---- begin button ----
<rect> ---- rectangle ----
0
32
20
49
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Gyro CounterClockwise
<endo> ---- end button
<cw_> ---- Clockwise Switch ----
<bgno> ---- begin button ----
<rect> ---- rectangle ----
22
32
43
49
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Gyro Clockwise
<endo> ---- end button
<sfx_> ---- switch sfx ----
switchup.wav
switchdn.wav
<help> ---- popup help ----
Gyro Slaving Controls
<endo> ==== END GAUGE ENTRY ====

```

gyro Directional Gyro Needle Gauge

Class: CDirectionalGyroGauge

Base Class: CNeedleGauge

Description:

Use this gauge type for Gyro compasses. Must connect to the Gyro pneumatic subsystem, 'GYRO'

Read Tags

Signature	Description	Type	Format
apbg	Autopilot bug	ART	Bug file (PBG)
		INT	X offset
		INT	Y offset
		ID_TYPE	Bug tag
apbk	Autopilot bug knob	ART	Bug knob file (PBG)
		INT	Top-left X location
		INT	Top-left Y location
		INT	Bottom-right X location
		INT	Bottom-right Y location
knob	Gyro correction knob	ART	Knob file (PBG)
		INT	Top-left X location
		INT	Top-left Y location
		INT	Bottom-right X location
		INT	Bottom-right Y location
bias	Bias tag (Use this if gauge is tied to FCS)	ID_TYPE	Gyro bias tag?

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
gyro
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  dgyr
  <size> ---- x,y,xsize,ysize ----
  313
  287
  90
  90
  <mesg>
  <bgno>
    <conn>
    gyro
    <dtag>
    yaw_
  <endo>
  <over> ---- overlay image ----
  GYROOVER.PBM
  2
  2
  <cntr> ---- needle center x,y ----
  45
  45
  <rang> ---- angle range ----
  0.0
  360.0
  <nedl> ---- bitmap needle ----
  gyro
```



```
<bgn> ===== BEGIN BITMAP NEEDLE =====
    <file> ---- needle file ----
        DGYRO.NDL
<end> ===== END BITMAP NEEDLE =====
<knob> ---- gyro correction knob ----
dgknob.pbg
-4
76
15
94
<apbg> ---- autopilot bug ----
apbug.pbg
2
2
_Bug
<apkb> ---- autopilot bug knob ----
skyhapkb.pbg
75
76
94
94
<bias> ---- bias tag ----
gyro
<crsr> ---- default cursor ----
mpfinger.csr
<help> ---- popup help ----
Directional Gyro
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhdgym.raw
<end> ===== END GAUGE ENTRY =====
```

happ Walker Autopilot Panel

Class: CWalkerAPPanel

Base Class: CGauge

Description:

The Walker autopilot is part of the avionics suite implemented in the TRI Peregrine 800XP jet.

Read Tags

Signature	Description	Type	Format
left	Bank Left Button	OBJECT	Click Area
right	Bank Right Button	OBJECT	Click Area
up__	Pitch Up Button	OBJECT	Click Area
down	Pitch Down Button	OBJECT	Click Area
xfer	Transfer Button	OBJECT	Click Area
turb	Turbulence Button	OBJECT	Click Area
ydon	Yaw Damper On Button	OBJECT	Click Area
ydoF	Yaw Damper Off Button	OBJECT	Click Area
apon	Autopilot On Button	OBJECT	Click Area
apof	Autopilot Off Button	OBJECT	Click Area
enga	Engage Slider	ART	PBG filename
		ART	PBG filename
knob	Bank Rate Knob	ART	PBG filename
		ART	PBG filename

Example (TRI40.PNL)

```
<gage> ===== GAUGE ENTRY =====
happ
<bgno> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
  happ
  <size> ---- x,y,xsize,ysize ----
  659
  380
  127
  51
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data connection ----
    MrAp
    <user> ---- user data ----
    HARDWARE,OTHER
    <user> ---- user data ----
    UNIT,1
    <dtag> ---- user data ----
    apOn
  <endo>
  <left> ---- bank left ----
  <bgno> ---- begin button ----
    <rect> ---- rectangle ----
    1
    3
    17
    40
    <curs> ---- cursor ----
    mpfliplt.csr
    <help> ---- popup help ----
```



```

Bank Left
<endo> ---- end button
<rght> ---- bank right ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    29
    3
    45
    40
    < curs> ---- cursor ----
    mpfliprt.csr
    <help> ---- popup help ----
    Bank Right
<endo> ---- end button
<up__> ---- rate up ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    18
    10
    28
    24
    < curs> ---- cursor ----
    mpflipup.csr
    <help> ---- popup help ----
    Rate Up
<endo> ---- end button
<down> ---- rate down ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    18
    25
    28
    40
    < curs> ---- cursor ----
    mpflipdn.csr
    <help> ---- popup help ----
    Rate Down
<endo> ---- end button
<xfer> ---- Xfer ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    48
    5
    68
    19
    <bmap> ---- bitmap ----
    rhapxfer.pbg
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Autopilot Transfer
<endo> ---- end button
<turb> ---- Turbulence ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    48
    30
    68
    45
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Turbulence
<endo> ---- end button
<ydon> ---- Yaw Damper On ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
    79
    3
    97
    25

```

```

        <curs> ---- cursor ----
        mpflipup.csr
        <help> ---- popup help ----
        Yaw Damper Engage
    <endo> ---- end button
    <ydoof> ---- Yaw Damper Off ----
    <bgno> ---- begin button ----
        <rect> ---- rectangle ----
        79
        26
        97
        47
        <curs> ---- cursor ----
        mpflipdn.csr
        <help> ---- popup help ----
        Yaw Damper Disengage
    <endo> ---- end button
    <apon> ---- AP On ----
    <bgno> ---- begin button ----
        <rect> ---- rectangle ----
        98
        3
        116
        25
        <curs> ---- cursor ----
        mpflipup.csr
        <help> ---- popup help ----
        Autopilot Engage
    <endo> ---- end button
    <apof> ---- Autopilot Off ----
    <bgno> ---- begin button ----
        <rect> ---- rectangle ----
        98
        26
        116
        47
        <curs> ---- cursor ----
        mpflipdn.csr
        <help> ---- popup help ----
        Autopilot Disengage
    <endo> ---- end button
    <enga> ---- engage sliders ----
    rhapyden.pbg
    rhapapen.pbg
    <knob> ---- bank/rate knob ----
    rhapbnk.pbg
    rhaptrm.pbg
    <lite> ---- light ----
    plfl
    <help> ---- popup help ----
    Autopilot Controls
<endo> ==== END GAUGE ENTRY ====

```

hbal Horizontal Ball Gauge

Class: CHorizontalBallGauge

Base Class: CBitmapGauge

Description:

This gauge type implements a simple attitude indicator horizon ball gauge, typically used as a backup indicator to the primary EFIS display panels. There are no gauge-specific tags.

Read Tags

Signature	Description	Type	Format

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
hbal
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    hbal
    <size> ---- x,y,xsize,ysize ----
    536
    253
    74
    75
    <over> ---- overlay ----
    rhattiov.pbm
    -9
    -8
    <lite> ---- light ----
    plfl
    <help> ---- popup help ----
    Artificial Horizon Ball
<endo> ==== END GAUGE ENTRY ====
```



hobb Hobbs Meter Gauge

Class: CHobbsMeterGauge

Base Class: CGauge

Description:

This common gauge type implements a roller-type Hobbs Meter which indicates the number of hours that the aircraft engine has been running. The individual digit artwork is specified by the <blak> and <whit> tags which each specify a PBG bitmap containg all digits 0-9. The black digits are used for all whole numbers, and the white digit indicates fractional hours.

Incidentally, the hobbs meter setting for each aircraft is recorded between sessions in the aircraft .HST file.

Read Tags

Signature	Description	Type	Format
blak	Black Digits Artwork	ART	PBG filename
whit	White Digits Artwork	ART	PBG filename

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
hobb
<bgn> ==== BEGIN GAUGE ====
  <unid> ---- unique id ----
    hob2
  <size> ---- placement and size ----
    1233
    259
    40
    11
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- info tag ----
      hobs
    <user> ---- user data ----
      HARDWARE, GAUGE
  <endo>
  <blak> ---- black digits ----
    skyhmlb.pbg
  <whit> ---- white digits ----
    skyhmlw.pbm
  <help> ---- help string ----
    Hobbs Meter
<endo> ==== END GAUGE ====
```



hori Artificial Horizon Gauge

Class: CHorizonGauge

Base Class: CBitmapGauge

Description:

This common gauge type implements a traditional "steam gauge" version of the attitude (artificial horizon) indicator.

Read Tags

Signature	Description	Type	Format
foot	Football artwork (football-shaped horizon bar)	ART	Football file (PBG)
		INT	X coordinate
		INT	Y coordinate
horz	Horizon artwork (Horizon needle or small airplane symbol)	ART	Horizon file (PBG)
		INT	X coordinate
		INT	Y coordinate
hoff	Horizon offset (Default horizon needle offset in Y axis)	INT	Pixels (default 0)
irim	Inner rim artwork (Ring with bank markings)	ART	Inner rim file (PBG)
		INT	X coordinate
		INT	Y coordinate
orim	Outer rim artwork (Sometimes a separate piece of art is needed for an outer ring)	ART	Outer rim file (PBM)
		INT	X coordinate
		INT	Y coordinate
fron	Front plate artwork (Front face plate for the instrument)	ART	Front plate file (PBM)
		INT	X coordinate
		INT	Y coordinate
pixd	Pixels per degree (football) (Use this to calibrate the pitch reading)	REAL	
moff	Maximum drawing offset	INT	Pixels
knob	Horizon needle adjustment knob	OBJECT	Knob
flag	Gyro flag name?	ART	Flag file (PBM)
pich	Pitch message	OBJECT	Message
roll	Roll message	OBJECT	Message

Example (FLYHWK00.PNL)

```
<gage> ===== GAUGE ENTRY =====
hori
<bgno> ===== BEGIN GAUGE ENTRY =====
    <unid> ----- unique id -----
    hori
    <size> ----- x,y,xsize,ysize -----
    312
    170
    90
    90
    <mesg>
    <bgno>
        <conn>
        atti
        <dtag>
        atti
    <endo>
    <foot> ----- football filename -----
    ahoriz_b.pbg
    2
    3
    <horz> ----- horizon needle filename -----
    ahoriz_c.pbm
    2
    3
    <irim> ----- inner rim filename -----
    ahoriz_d.pbg
    2
    3
    <orim> ----- outer rim filename -----
    ahoriz_f.pbm
    2
    3
    <fron> ----- front plate filename -----
    skyhatti.pbm
    0
    0
    <pixd> ----- pixels per degree -----
    0.6
    <moff> ----- maximum pixel offset -----
    23
    <help> ----- popup help -----
    Attitude Indicator
    <lite> ----- light flag -----
    gage
    <mask> ----- light mask -----
    skyhattm.raw
    <knob> ----- horizon cross knob -----
    <bgno> ----- begin click area -----
        <bmap> ----- bitmap -----
        skyhknob.pbg
        <rect> ----- click area rect -----
        36
        71
        54
        89
        <curs> ----- cursor -----
        mpfinger.csr
        <help> ----- popup help -----
        Horizon Needle Adjustment
    <endo> ----- end click area -----
<endo> ===== END GAUGE ENTRY =====
```



hsat Walker Temperature/Airspeed Gauge

Class: CWalkerTemperatureAirSpeedGauge

Base Class: CGauge

Description:

The Walker Temp/Airspeed gauge is a part of the avionics suite implemented in the TRI Peregrine 800XP jet aircraft.

Temperature display can be toggled between static and total air temperature by clicking on either the <tat_> or <sat_> click areas. The airspeed display can be toggled between indicated and true airspeed by clicking on the <tas_> click area.

Read Tags

Signature	Description	Type	Format
tat_	True Air Temp Button	OBJECT	Click Area
sat_	Static Air Temp Button	OBJECT	Click Area
tas_	True Airspeed Button	OBJECT	Click Area

Example (TRI40.PNL)

```
<gage> ===== GAUGE ENTRY =====
hsat
<bgn> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
    tasd
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- data tag ----
    tasd
    <user> ---- user data ----
    HARDWARE, OTHER
  <end>
  <size> ---- x,y,xsize,ysize ----
  754
  161
  90
  35
  <tat_> ---- TAT button ----
  <bgn> ---- begin ----
    <rect> ---- rect ----
    1
    12
    9
    25
    <curr> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Total Air Temperature
  <end> ---- end ----
  <sat_> ---- SAT display ----
  <bgn> ---- begin ----
    <rect> ---- rect ----
    16
    13
    40
    23
    <help> ---- help ----
```



```
        Air Temperature
<endo> ---- end ----
<tas_> ---- TAS display ----
<bgno> ---- begin ----
        <rect> ---- rect ----
        55
        13
        79
        23
        <help> ---- help ----
        True Airspeed
<endo> ---- end ----
<help> ---- popup help ----
        Air Temperature/Airspeed Display
<endo> ==== END GAUGE ENTRY ====
```

hsi_ Horizontal Situation Indicator Gauge

Class: CHorizontalSituationIndicator

Base Class: CBitmapGauge

Description:

This gauge type implements a Horizontal Situation Indicator.

Read Tags

Signature	Description	Type	Format
comp	Compass Plate	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
cour	Course Needle	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
devi	Deviation Needle	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
devb	Deviation Bar	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
apbg	Autopilot Bug	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
to__	TO Flag	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
from	FROM Flag	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
glid	Glideslope Needle	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
knob	Course Knob	OBJECT	Knob
apkb	Autopilot Knob	OBJECT	Knob
obsm	OBS Message	OBJECT	Message
bugm	AP Bug Message	OBJECT	Message
fcs_	FCS Message	OBJECT	Message

Example (MALI00.PNL)

```
<gage> ==== GAUGE ENTRY ====
hsi_
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    hsi_
  <size> ---- size and location ----
    363
    255
    90
    90
  <mesg>
  <bgn>
    <conn>
```



```

        hsi1
        <dtag>
        yaw_
<endo>
<over> ---- overlay image ----
maligyro.pbm
-12
-10
<cntr> ---- needle center x,y ----
45
46
<rang> ---- angle range ----
0.0
360.0
<comp> ---- compass plate ----
dgyro.pbg
2
3
<cour> ---- course needle ----
hsiouter.pbg
4
4
<devi> ---- deviation needle ----
hsiinner.pbg
4
4
<devb> ---- deviation bar ----
hsierror.pbg
4
4
<apbg> ---- autopilot bug ----
apbug.pbg
2
1
<to__> ---- to indicator ----
hsito.pbg
2
2
<from> ---- from indicator ----
hsifrom.pbg
2
2
<glid> ---- glideslope needle ----
hsigs90.pbm
0
39
<knob> ---- course knob ----
<bgno> ---- begin area ----
    <bmap> ---- bitmap ----
    dgknob.pbg
    <rect> ---- click rect ----
    -4
    78
    15
    96
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Course Selector
<endo> ---- end area ----
<apkb> ---- course knob ----
<bgno> ---- begin area ----
    <bmap> ---- bitmap ----
    skyhapkb.pbg
    <rect> ---- click rect ----
    75
    78
    94
    96
    <curs> ---- cursor ----
    mpfinger.csr

```

```

        <help> ---- popup help ----
        Heading Bug
<endo> ---- end area ----
<obsm> ---- course heading message ----
<bgno> ---- begin message ----
        <conn> ---- group ----
        nav1
        <user> ---- user data ----
        HARDWARE, GAUGE
        <dtag> ---- user data ----
        obs_
<endo> ---- end message ----
<bugm> ---- heading bug message ----
        <conn> ---- group ----
        K150
        <user> ---- user data ----
        HARDWARE, OTHER
        <dtag> ---- group ----
        _Bug
<bgno> ---- begin message ----
<endo> ---- end message ----
<fcs_> ---- FCS switch ----
<bgno> ---- begin message ----
        <conn> ---- group ----
        AXIS
        <user> ---- user data ----
        HARDWARE, OTHER
        <dtag> ---- user data ----
        _FCS
<endo> ---- end message ----
<help> ---- popup help ----
Horizontal Situation Indicator
<mask> ---- light mask ----
pmmask06.raw
<lite> ---- light source ----
avio
<endo> ===== END GAUGE ENTRY =====

```

hspt Momentary Hot Spot Gauge

Class: CMomentaryHotSpotGauge

Base Class: CBitmapGauge

Description:

The HotSpot gauge type is used to implement a simple clickable switch area within the panel that does not have any associated graphics. For example, very small pushbuttons might be defined with this gauge type if there is no visible difference between the button being in the "up" and "down" states.

The state is always momentary-contact, i.e. the switch state is Active (On) as long as the user has the mouse button depressed on the gauge region; as soon as the mouse button is released, the switch goes back to the Inactive (Off) state.

There are no gauge-specific data tags.

Read Tags

Signature	Description	Type	Format

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
hspt
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    pats
    <mesg> -- Subsystem Message --
    <bgno>
        <conn>
        test
        <user>
        HARDWARE, SWITCH
    <endo>
    <size> ---- x,y,xsize,ysize ----
    916
    339
    17
    17
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Annunciator Test
<endo> ==== END GAUGE ENTRY ====
```



indi Indicator Gauge

Class: CIndicatorGauge

Base Class: CGauge

Description:

The basic indicator gauge is typically used for such things as trim indicators. The <mesg> data connection must return a REAL data value which is mapped to the indicator gauge animation frames in the PBG file specified by the <bmap> tag. By default, the expected data range is between 0.0 and 1.0 (this can be overridden using the <lval> and <hval> tags). If for example the indicator artwork had 10 distinct sub-images within the PBG file, then frame 0 would be used for values between 0.0 and 0.1; frame 1 for values between 0.1 and 0.2, etc. up to frame 9 for values between 0.9 and 1.0.

Read Tags

Signature	Description	Type	Format
lval	Low Value	REAL	Low limit of indication
hval	High Value	REAL	High limit of indication
bmap	Indicator Artwork	ART	PBG filename

Example (TRI41.PNL)

```
<gage> ==== GAUGE ENTRY ====
indi
<bgn> ==== BEGIN OBJECT ====
    <unid>
    etri
    <mesg> -- Subsystem Message --
    <bgn>
        <conn> ---- Elevator Trim ----
        etrm
        <dtag> ---- data tag ----
        unit
    <endo>
    <lval>
    1.0
    <hval>
    -1.0
    <size> ---- size and pos ----
    278
    76
    16
    102
    <bmap> ---- bitmaps ----
    rheltrmi.pbg
    <help> ---- popup help ----
    Elevator Trim Indicator
<endo> ==== END OBJECT ====
```

k155 Bendix-King KX-155 Nav/Comm Radio

Class: CBKNavCommKX155Gauge

Base Class: CBitmapGauge

Read Tags

Signature	Description	Type	Format
radi	Radio Identifier	ID_TYPE	Radio subsystem ID
		INT	Subsystem unit number
ca01	COM Active Whole MHz	OBJECT	Click Area
ca02	COM Active Fract MHz	OBJECT	Click Area
ca03	COM Standby Whole MHz	OBJECT	Click Area
ca04	COM Standby Fract MHz	OBJECT	Click Area
ca05	NAV Active Whole MHz	OBJECT	Click Area
ca06	NAV Active Fract MHz	OBJECT	Click Area
ca07	NAV Standby Whole MHz	OBJECT	Click Area
ca08	NAV Standby Fract MHz	OBJECT	Click Area
ca09	COM Tuning Knob	OBJECT	Click Area
ca10	NAV Tuning Knob	OBJECT	Click Area
ca11	COM Channel Button	OBJECT	Click Area
ca12	NAV Mode Button	OBJECT	Click Area
ca13	COM Transfer Button	OBJECT	Click Area
ca14	NAV Transfer Button	OBJECT	Click Area
ca15	COM Volume Knob	OBJECT	Click Area
ca16	NAV Volume Knob	OBJECT	Click Area
ca17	OBS Digit 1	OBJECT	Click Area
ca18	OBS Digit 2	OBJECT	Click Area
ca19	OBS Digit 3	OBJECT	Click Area
ca20	Timer Minutes	OBJECT	Click Area
ca21	Timer Seconds	OBJECT	Click Area

Example

Due to the length and repetitive nature of this gauge specification, the example is not copied here. Examples can be found in many of the main panel files, such as FLYHWK00.PNL, and MALI00.PNL



ka14 Bendix-King KAP-140 Autopilot

Class: CBKAP140Gauge

Base Class: CBitmapGauge

Description:

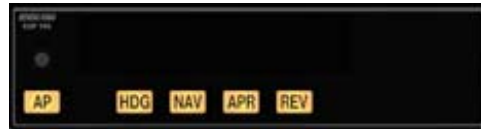
This gauge type implements the Bendix-King KAP-140 Autopilot panel.
This is a simple single-axis autopilot.

Read Tags

Signature	Description	Type	Format
ca01	AP Engage/Disengage	OBJECT	Click Area
ca02	HDG Button	OBJECT	Click Area
ca03	NAV Button	OBJECT	Click Area
ca04	APR Button	OBJECT	Click Area
ca05	REV Button	OBJECT	Click Area
ca06	AP Engaged Light	OBJECT	Click Area

Example

Due to the length and repetitive nature of this gauge specification, the example is not copied here. An example can be found in the default TRI Flyhawk main panel, FLYHWK00.PNL.



kafe Beech King Air Fire Extinguisher

Class: CKAFireExtinguisherButton

Base Class: CGauge

Description:

Read Tags

Signature	Description	Type	Format
fire	Fire Light	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
ok__	Fire Light	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
d__	Discharge	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
sfx_	Sound Effect	SOUND	WAV filename
curs	Cursor	CURSOR	CSR filename

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
kafe
<bgn0> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  lfeb
  <size> ---- x,y,xsize,ysize ----
  580
  361
  39
  36
  <mesg> ---- message ----
  <bgn0> ---- begin ----
    <conn> ---- group ----
    LFir
    <dtag>
    fire
    <user>
    ENGINE,1
  <endo> ---- end ----
  <fire> ---- fire light ----
  kafireex.pbg
  1
  1
  <ok__> ---- OK light ----
  kafireok.pbg
  19
  17
  <d__> ---- Discharge light ----
  kafired.pbg
  1
  17
  <sfx_> ---- switch sfx ----
  switchdn.wav
  <curs> ---- cursor ----
  mpfinger.csr
  <help> ---- popup help ----
  Left Engine Fire Extinguish
```



<endo> ==== END GAUGE ENTRY ====

kf15 Bendix-King KFC-150 Autopilot

Class: CBKKFC150Gauge

Base Class: CBitmapGauge

Description:

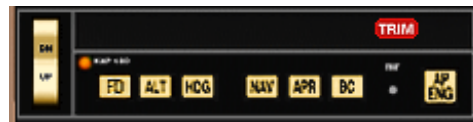
This gauge type implements the Bendix-King KFC-150 Autopilot panel. This is a two-axis autopilot that is capable of controlling both heading and altitude.

Read Tags

Signature	Description	Type	Format
ca01	AP Engage/Disengage	OBJECT	Click Area
ca02	HDG Button	OBJECT	Click Area
ca03	NAV Button	OBJECT	Click Area
ca04	APR Button	OBJECT	Click Area
ca05	BC Button	OBJECT	Click Area
ca06	Engage Light	OBJECT	Click Area
ca07	ALT Button	OBJECT	Click Area
ca08	VS Rocker Down	OBJECT	Click Area
ca09	VS Rocker Up	OBJECT	Click Area
ca10	FD Button	OBJECT	Click Area
ca11	Test Button	OBJECT	Click Area
fd__	FD Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset
alt_	ALT Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset
hdg_	HDG Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset
gs__	GS Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset
nav_	NAV Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset
apr_	APR Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset
bc_	BC Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset
trim	Trim Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset
ap__	AP Annunciator	ART	PBM filename
		INT	X pixel offset
		INT	Y pixel offset

Example (MALI00.PNL)

Due to the length and repetitive nature of this gauge specification, the example is not copied here. An example can be found in the default TRI Saharha main panel, MALI00.PNL.



kf20 Bendix-King KFC-200 Autopilot

Class: CBKKFC200Gauge

Base Class: CBitmapGauge

Description:

Read Tags

Signature	Description	Type	Format

Example

khfr Collins HF Radio Gauge

Class: CCollinsHFRadioGauge

Base Class: CBitmapGauge

Description:

This gauge type implements the Collins High-Frequency Radio.

Read Tags

Signature	Description	Type	Format
ft13	Use 13-point LED Font	NO_TYPE	
chan	Channel	OBJECT	Click Area
mhz_	Frequency MHz	OBJECT	Click Area
khz_	Frequency KHz	OBJECT	Click Area
tx__	TX Indicator	OBJECT	Click Area
dash	Dash Indicator	OBJECT	Click Area
volu	Volume Knob	OBJECT	Knob
tune	Tune Knob	OBJECT	Knob
stor	STO Button	OBJECT	Click Area
lsb_	LSB (Lower Sideband) Mode	OBJECT	Click Area
usb_	USB (Upper Sideband) Mode	OBJECT	Click Area
am__	AM Mode	OBJECT	Click Area
tel_	Tel Mode	OBJECT	Click Area

Example (TRI44.PNL)

```
<gage> ==== GAUGE ENTRY ====
khfr
<bgno> ==== BEGIN GAUGE ENTRY ====
  <size> ---- x,y,xsize,ysize ----
    469
    262
    93
    102
  <mesg> ---- message ----
  <bgno> ---- begin ----
    <conn> ---- provider ----
      hifr
    <user> ---- hardware ----
      HARDWARE,RADIO
    <user> ---- unit ----
      UNIT,1
  <endo>
  <ft13>
  <chan> ---- channel ----
  <bgno> ---- begin ----
    <rect>
      11
      18
      28
      30
    <curs>
      mpfinger.csr
    <help>
      Channel
  <endo> ---- end ----
  <mhz_> ---- MHz ----
```



```

<bgn> ---- begin ----
    <rect>
    44
    18
    63
    30
    <curr>
    mpfinger.csr
    <help>
    Frequency (MHz)
<end> ---- end ----
<khz_> ---- KHz ----
<bgn> ---- begin ----
    <rect>
    24
    34
    63
    46
    <curr>
    mpfinger.csr
    <help>
    Frequency (KHz)
<end> ---- end ----
<tx_> ---- TX ----
<bgn> ---- begin ----
    <rect>
    66
    18
    70
    30
    <curr>
    mpfinger.csr
<end> ---- end ----
<dash> ---- "-" ----
<bgn> ---- begin ----
    <rect>
    35
    18
    42
    30
    <curr>
    mpfinger.csr
<end> ---- end ----
<volu> ---- Volume ----
<bgn> ---- begin ----
    <rect>
    18
    79
    33
    96
    <curr>
    mpfinger.csr
    <bmap>
    rhlsvol.pbg
    <help>
    Volume
<end> ---- end ----
<tune> ---- tune ----
<bgn> ---- begin ----
    <rect>
    45
    63
    87
    101
    <curr>
    mpfinger.csr
    <bmap>
    rhlshf.pbg
    <help>
    Mode/Tune
<end> ---- end ----

```

```

<stor> ---- ST0 ----
<bgn> ---- begin ----
      <rect>
      82
      41
      87
      53
      <curr>
      mpfinger.csr
      <help>
      Store
<end> ---- end ----
<lsb_> ---- LSB ----
<bgn> ---- begin ----
      <rect>
      38
      70
      49
      75
      <curr>
      mpfinger.csr
      <help>
      Mode (LSB)
<end> ---- end ----
<usb_> ---- USB ----
<bgn> ---- begin ----
      <rect>
      48
      60
      59
      65
      <curr>
      mpfinger.csr
      <help>
      Mode (USB)
<end> ---- end ----
<am_> ---- AM ----
<bgn> ---- begin ----
      <rect>
      70
      59
      79
      65
      <curr>
      mpfinger.csr
      <help>
      Mode (AM)
<end> ---- end ----
<tel_> ---- MHz ----
<bgn> ---- begin ----
      <rect>
      82
      67
      92
      76
      <curr>
      mpfinger.csr
      <help>
      Mode (Tel)
<end> ---- end ----
<help> ---- popup help ----
King KHF-950
<end> ===== END GAUGE ENTRY =====

```

kl89 Bendix-King KLN-89 GPS

Class: CBKGPSKLN89Gauge

Base Class: CBitmapGauge

Read Tags

Signature	Description	Type	Format
radi	Radio Subsystem	ID_TYPE	GPS subsystem ID
ca01	MSG Button	OBJECT	Click Area
ca02	OBS Button	OBJECT	Click Area
ca03	ALT Button	OBJECT	Click Area
ca04	NRST Button	OBJECT	Click Area
ca05	D->TO Button	OBJECT	Click Area
ca06	CLR Button	OBJECT	Click Area
ca07	ENT Button	OBJECT	Click Area
ca08	CRSR Button	OBJECT	Click Area
ca09	Knob	OBJECT	Click Area
ca10	APT Function	OBJECT	Click Area
ca11	VOR Function	OBJECT	Click Area
ca12	NDB Function	OBJECT	Click Area
ca13	INT Function	OBJECT	Click Area
ca14	USR Function	OBJECT	Click Area
ca15	ACT Function	OBJECT	Click Area
ca16	NAV Function	OBJECT	Click Area
ca17	FPL Function	OBJECT	Click Area
ca18	CAL Function	OBJECT	Click Area
ca19	SET Function	OBJECT	Click Area
ca20	OTH Function	OBJECT	Click Area
ca21	LED	OBJECT	Click Area
ca22	Power Knob	OBJECT	Knob

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
kl89
<bgn> ==== BEGIN OBJECT ====
  <unid> ---- unique id ----
  gpsg
  <size> ---- x,y,xsize,ysize ----
  671
  168
  226
  66
  <radi> ---- radio identifier ----
  gpsr
  1
  <ca01> ---- MSG button ----
  <bgn> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt34.pbg
    <rect> ---- click area rect ----
    42
    53
    61
    65
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
```



```

        Message Button
<endo> ---- end click area ----
<ca02> ---- OBS button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt35.pbg
        <rect> ---- click area rect ----
        62
        53
        81
        65
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        OBS Button
<endo> ---- end click area ----
<ca03> ---- ALT button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt36.pbg
        <rect> ---- click area rect ----
        82
        53
        101
        65
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Altitude Button
<endo> ---- end click area ----
<ca04> ---- NRST button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt37.pbg
        <rect> ---- click area rect ----
        102
        53
        121
        65
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Nearest Button
<endo> ---- end click area ----
<ca05> ---- ARROW button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt38.pbg
        <rect> ---- click area rect ----
        121
        53
        140
        65
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Direct-To Button
<endo> ---- end click area ----
<ca06> ---- CLR button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt39.pbg
        <rect> ---- click area rect ----
        141
        53
        160
        65
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Clear Button

```

```

<endo> ---- end click area ----
<ca07> ---- ENT button ----
<bgn0> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt40.pbg
    <rect> ---- click area rect ----
    161
    53
    180
    65
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Enter Button
<endo> ---- end click area ----
<ca08> ---- CRSR button ----
<bgn0> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt41.pbg
    <rect> ---- click area rect ----
    200
    12
    219
    24
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Cursor Button
<endo> ---- end click area ----
<ca09> ---- Knob ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    194
    31
    218
    56
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Knob
<endo> ---- end click area ----
<ca10> ---- APT ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    45
    45
    56
    52
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Airport Pages
<endo> ---- end click area ----
<ca11> ---- VOR ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    57
    45
    68
    52
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    VOR Pages
<endo> ---- end click area ----
<ca12> ---- NDB ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    69
    45
    80

```

```

52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
NDB Pages
<endo> ---- end click area ----
<ca13> ---- INT ----
<bgn> ---- begin click area ----
<rect> ---- click area rect ----
81
45
92
52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Intersection Pages
<endo> ---- end click area ----
<ca14> ---- USR ----
<bgn> ---- begin click area ----
<rect> ---- click area rect ----
93
45
104
52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
User Pages
<endo> ---- end click area ----
<ca15> ---- ACT ----
<bgn> ---- begin click area ----
<rect> ---- click area rect ----
105
45
117
52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Active Page
<endo> ---- end click area ----
<ca16> ---- NAV ----
<bgn> ---- begin click area ----
<rect> ---- click area rect ----
118
45
129
52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
NAV Pages
<endo> ---- end click area ----
<ca17> ---- FPL ----
<bgn> ---- begin click area ----
<rect> ---- click area rect ----
130
45
141
52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Flight Plan Pages
<endo> ---- end click area ----
<ca18> ---- CAL ----
<bgn> ---- begin click area ----
<rect> ---- click area rect ----
142
45

```

```

153
52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Calculator Pages
<endo> ---- end click area ----
<ca19> ---- SET ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
154
45
165
52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
SET Pages
<endo> ---- end click area ----
<ca20> ---- OTH ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
166
45
177
52
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Other Pages
<endo> ---- end click area ----
<ca21> ---- LED ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
36
6
184
44
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
LED Display
<endo> ---- end click area ----
<ca22> ---- Power ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
5
41
34
66
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Power
<bmap> ---- bitmap ----
kln89pwr.pbg
<endo> ---- end click area ----
<lite> ---- light flag ----
radi
<help> ---- popup help ----
Bendix/King KLN89 GPS radio
<endo> ==== END OBJECT ====

```

kma2 Bendix-King KMA-26 Audio Panel

Class: CBKAudioKMA26Gauge

Base Class: CBitmapGauge

Description:

Read Tags

Signature	Description	Type	Format
radi	Audio Radio Subsystem	ID_TYPE	Subsystem ID
ca01	COM1 Button	OBJECT	Click Area
ca02	COM2 Button	OBJECT	Click Area
ca03	COM2 Button	OBJECT	Click Area
ca04	NAV1 Button	OBJECT	Click Area
ca05	NAV2 Button	OBJECT	Click Area
ca06	MKR Button	OBJECT	Click Area
ca07	DME Button	OBJECT	Click Area
ca08	ADF Button	OBJECT	Click Area
ca09	AUX Button	OBJECT	Click Area
ca10	MONI Button	OBJECT	Click Area
ca11	Outer Marker	OBJECT	Click Area
ca12	Middle Marker	OBJECT	Click Area
ca13	Inner Marker	OBJECT	Click Area
ca14	Volume Knob	OBJECT	Click Area
ca15	Microphone Mode	OBJECT	Click Area

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
kma2
<bgno> ==== BEGIN OBJECT ====
  <unid> ---- unique id ----
    kma2
  <size> ---- x,y,xsize,ysize ----
    671
    117
    226
    44
  <radi> ---- radio identifier ----
    adio
    1
  <ca01> ---- COM1 button ----
  <bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
      skyhbt24.pbg
    <rect> ---- click area rect ----
      83
      3
      99
      18
    <curs> ---- cursor ----
      mpfinger.csr
    <help> ---- popup help ----
      COM1 Audio Select
  <endo> ---- end click area ----
  <ca02> ---- COM2 button ----
  <bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
      skyhbt25.pbg
```



```

    <rect> ---- click area rect ----
    102
    3
    118
    18
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    COM2 Audio Select
<endo> ---- end click area ----
<ca03> ---- COM3 button ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt26.pbg
    <rect> ---- click area rect ----
    121
    3
    137
    18
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    COM3 Audio Select
<endo> ---- end click area ----
<ca04> ---- NAV1 button ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt27.pbg
    <rect> ---- click area rect ----
    141
    3
    157
    18
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    NAV1 Audio Select
<endo> ---- end click area ----
<ca05> ---- NAV2 button ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt28.pbg
    <rect> ---- click area rect ----
    160
    3
    176
    18
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    NAV2 Audio Select
<endo> ---- end click area ----
<ca06> ---- MKR button ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt29.pbg
    <rect> ---- click area rect ----
    83
    24
    99
    39
    < curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Marker Audio Select
<endo> ---- end click area ----
<ca07> ---- DME button ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt30.pbg
    <rect> ---- click area rect ----

```

```

102
24
118
39
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
DME Audio Select
<endo> ---- end click area ----
<ca08> ---- ADF button ----
<bgno> ---- begin click area ----
<bmap> ---- bitmap ----
skyhbt31.pbg
<rect> ---- click area rect ----
122
24
138
39
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
ADF Audio Select
<endo> ---- end click area ----
<ca09> ---- AUX button ----
<bgno> ---- begin click area ----
<bmap> ---- bitmap ----
skyhbt32.pbg
<rect> ---- click area rect ----
141
24
157
39
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Auxillary Audio Select
<endo> ---- end click area ----
<ca10> ---- MONI button ----
<bgno> ---- begin click area ----
<bmap> ---- bitmap ----
skyhbt33.pbg
<rect> ---- click area rect ----
160
24
178
39
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Monitor Audio Select
<endo> ---- end click area ----
<ca11> ---- Outer indicator ----
<bgno> ---- begin click area ----
<bmap> ---- bitmap ----
skyhoutr.pbg
<rect> ---- click area rect ----
9
5
21
16
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Outer Marker Indicator
<endo> ---- end click area ----
<ca12> ---- Middle indicator ----
<bgno> ---- begin click area ----
<bmap> ---- bitmap ----
skyhmidl.pbg
<rect> ---- click area rect ----
9

```

```

16
21
27
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Middle Marker Indicator
<endo> ---- end click area ----
<ca13> ---- Outer indicator ----
<bgn0> ---- begin click area ----
<bmap> ---- bitmap ----
skyhinnr.pbg
<rect> ---- click area rect ----
9
27
21
38
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Inner Marker Indicator
<endo> ---- end click area ----
<ca14> ---- Volume knob ----
<bgn0> ---- begin click area ----
<rect> ---- click area rect ----
55
12
75
30
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Volume Control
<endo> ---- end click area ----
<ca15> ---- Microphone Mode ----
<bgn0> ---- begin click area ----
<bmap> ---- bitmap ----
kma26kb1.pbg
<rect> ---- click area rect ----
181
4
220
38
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Microphone Mode
<endo> ---- end click area ----
<lite> ---- light flag ----
radi
<help> ---- popup help ----
Bendix/King KMA26 Audio Panel
<endo> ==== END OBJECT ====

```

kn53 King KN-53 Radio Gauge

Class: CBasicKingNavRadioGauge

Base Class: CNavRadioGauge

Description:

Unknown. No examples in any of the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

kn80 King KN-80 Radio Gauge

Class: CComplexKingNavRadioGauge

Base Class: CNavRadioGauge

Description:

Unkown. No examples in any of the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

knob Push/Pull Knob

Class: CPushPullKnobGauge

Base Class: CGauge

Description:

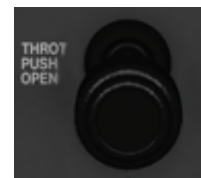
This gauge type implements a knob that uses a push/pull action to set an analog value, for example the throttle or mixture setting.

Read Tags

Signature	Description	Type	Format
bmap	Knob bitmap	ART	Knob art (PBG)
mesg	Subsystem message	OBJECT	Message
ytop	Y pixel value at top of knob	INT	
ybtm	Y pixel value at bottom of knob	INT	
valu	Click values	INT	Number of click areas
		INT	First state top pixel
		INT	First state bottom pixel
		REAL	First state value
		WORD	First state help text
		INT	Second state top pixel
		INT	Second state bottom pixel
		REAL	Second state value
		WORD	Second state help text
		...	continue for number of click areas specified
lowv	Low value	REAL	Default 0.0
high	High value	REAL	Default 1.0
poll	Polling message (specify this tag when ??)	NO_VALUE	

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
knob
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    thro
    <mesg>
    <bgno>
      <conn> ---- data tag ----
        THRO
        <dtag>
        thro
    <endo>
    <size> ---- x,y,xsize,ysize ----
    667
    569
    60
    97
    <bmap> ---- bitmap name ----
    skyhthro.pbg
    <poll> ---- polling ----
    <crsr> ---- default cursor ----
    mpslide.csr
```



```
      <help> ---- popup help ----  
      Engine Throttle  
<endo> ==== END GAUGE ENTRY ====
```

kr87 Bendix-King KR-87 ADF

Class: CBKADFKR87Gauge

Base Class: CBitmapGauge

Description:

Read Tags

Signature	Description	Type	Format
radi	ADF Radio Subsystem	ID_TYPE	Subsystem ID
ca01	Active Digit 1	OBJECT	Click Area
ca02	Active Digit 2	OBJECT	Click Area
ca03	Active Digit 3	OBJECT	Click Area
ca04	Active Digit 4	OBJECT	Click Area
ca05	Standby Digit 1	OBJECT	Click Area
ca06	Standby Digit 2	OBJECT	Click Area
ca07	Standby Digit 3	OBJECT	Click Area
ca08	Standby Digit 4	OBJECT	Click Area
ca11	ADF Button	OBJECT	Click Area
ca12	BFO Button	OBJECT	Click Area
ca13	FRQ Button	OBJECT	Click Area
ca14	FLT Button	OBJECT	Click Area
ca15	SET Button	OBJECT	Click Area

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
kr87
<bgn0> ==== BEGIN OBJECT ====
  <unid> ---- unique id ----
    kr87
  <size> ---- x,y,xsize,ysize ----
    670
    387
    227
    47
  <radi> ---- radio identifier ----
    adfr
    1
  <ca01> ---- active digit 1 ----
  <bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
      28
      7
      34
      16
    <curs> ---- cursor ----
      mpfinger.csr
    <help> ---- popup help ----
      ADF Active Frequency
  <endo> ---- end click area ----
  <ca02> ---- active digit 2 ----
  <bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
      35
      7
      41
      16
    <curs> ---- cursor ----
      mpfinger.csr
```



```

        <help> ---- popup help ----
        ADF Active Frequency
<endo> ---- end click area ----
<ca03> ---- active digit 3 ----
<bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        42
        7
        48
        16
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        ADF Active Frequency
<endo> ---- end click area ----
<ca04> ---- active digit 4 ----
<bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        49
        7
        55
        16
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        ADF Active Frequency
<endo> ---- end click area ----
<ca05> ---- standby digit 1 ----
<bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        93
        7
        99
        16
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        ADF Standby Frequency
<endo> ---- end click area ----
<ca06> ---- standby digit 2 ----
<bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        100
        7
        106
        16
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        ADF Standby Frequency
<endo> ---- end click area ----
<ca07> ---- standby digit 3 ----
<bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        107
        7
        113
        16
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        ADF Standby Frequency
<endo> ---- end click area ----
<ca08> ---- standby digit 4 ----
<bgn0> ---- begin click area ----
        <rect> ---- click area rect ----
        114
        7
        120
        16
        <curs> ---- cursor ----

```

```

        mpfinger.csr
        <help> ---- popup help ----
        ADF Standby Frequency
<endo> ---- end click area ----
<ca11> ---- ADF button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt03.pbg
        <rect> ---- click area rect ----
        32
        31
        51
        43
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        ADF/ANT Select
<endo> ---- end click area ----
<ca12> ---- BFO button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt04.pbg
        <rect> ---- click area rect ----
        56
        31
        75
        43
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        BFO Select
<endo> ---- end click area ----
<ca13> ---- FRQ button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt05.pbg
        <rect> ---- click area rect ----
        79
        31
        98
        43
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        ADF Frequency Transfer
<endo> ---- end click area ----
<ca14> ---- FLT button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt06.pbg
        <rect> ---- click area rect ----
        103
        31
        122
        43
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Flight/Elapsed Timer Select
<endo> ---- end click area ----
<ca15> ---- SET button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt07.pbg
        <rect> ---- click area rect ----
        127
        31
        146
        43
        <curs> ---- cursor ----
        mpfinger.csr

```

```
        <help> ---- popup help ----
        Set/Reset Timer
<endo> ---- end click area ----
<lite> ---- light flag ----
radi
<help> ---- popup help ----
Bendix/King KR87 ADF
<endo> ==== END OBJECT ====
```

kt76 Bendix-King KT-76 Transponder

Class: CBKXPDRKT76Gauge

Base Class: CBitmapGauge

Read Tags

Signature	Description	Type	Format
radi	XPDR Radio Subsystem	ID_TYPE	Subsystem ID
ca01	Digit 1	OBJECT	Click Area
ca02	Digit 2	OBJECT	Click Area
ca03	Digit 3	OBJECT	Click Area
ca04	Digit 4	OBJECT	Click Area
ca05	IDT Button	OBJECT	Click Area
ca06	Zero Button	OBJECT	Click Area
ca07	One Button	OBJECT	Click Area
ca08	Two Button	OBJECT	Click Area
ca09	Three Button	OBJECT	Click Area
ca10	Four Button	OBJECT	Click Area
ca11	Five Button	OBJECT	Click Area
ca12	Six Button	OBJECT	Click Area
ca13	Seven Button	OBJECT	Click Area
ca14	CLR Button	OBJECT	Click Area
ca15	VFR Button	OBJECT	Click Area
ca16	Mode Knob	OBJECT	Click Area

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
kt76
<bgno> ==== BEGIN OBJECT ====
  <unid> ---- unique id ----
    kt76
  <size> ---- x,y,xsize,ysize ----
    670
    441
    227
    54
  <radi> ---- radio identifier ----
    kt76
    1
  <ca01> ---- digit 1 ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
      113
      8
      119
      17
    <curs> ---- cursor ----
      mpfinger.csr
    <help> ---- popup help ----
      Transponder Code
  <endo> ---- end click area ----
  <ca02> ---- digit 2 ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
      120
      8
      126
      17
```



```

        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Transponder Code
<endo> ---- end click area ----
<ca03> ---- digit 3 ----
<bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        127
        8
        133
        17
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Transponder Code
<endo> ---- end click area ----
<ca04> ---- digit 4 ----
<bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        134
        8
        140
        17
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Transponder Code
<endo> ---- end click area ----
<ca05> ---- Ident button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt08.pbg
        <rect> ---- click area rect ----
        5
        11
        23
        23
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Ident Button
<endo> ---- end click area ----
<ca06> ---- Zero button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt09.pbg
        <rect> ---- click area rect ----
        4
        38
        22
        50
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Zero Button
<endo> ---- end click area ----
<ca07> ---- One button ----
<bgno> ---- begin click area ----
        <bmap> ---- bitmap ----
        skyhbt10.pbg
        <rect> ---- click area rect ----
        25
        38
        43
        50
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        One Button
<endo> ---- end click area ----

```

```

<ca08> ---- Two button ----
<bgno> ---- begin click area ----
      <bmap> ---- bitmap ----
      skyhbt11.pbg
      <rect> ---- click area rect ----
      46
      38
      64
      50
      <curs> ---- cursor ----
      mpfinger.csr
      <help> ---- popup help ----
      Two Button
<endo> ---- end click area ----
<ca09> ---- Three button ----
<bgno> ---- begin click area ----
      <bmap> ---- bitmap ----
      skyhbt12.pbg
      <rect> ---- click area rect ----
      67
      38
      85
      50
      <curs> ---- cursor ----
      mpfinger.csr
      <help> ---- popup help ----
      Three Button
<endo> ---- end click area ----
<ca10> ---- Four button ----
<bgno> ---- begin click area ----
      <bmap> ---- bitmap ----
      skyhbt13.pbg
      <rect> ---- click area rect ----
      88
      38
      106
      50
      <curs> ---- cursor ----
      mpfinger.csr
      <help> ---- popup help ----
      Four Button
<endo> ---- end click area ----
<ca11> ---- Five button ----
<bgno> ---- begin click area ----
      <bmap> ---- bitmap ----
      skyhbt14.pbg
      <rect> ---- click area rect ----
      109
      38
      127
      50
      <curs> ---- cursor ----
      mpfinger.csr
      <help> ---- popup help ----
      Five Button
<endo> ---- end click area ----
<ca12> ---- Six button ----
<bgno> ---- begin click area ----
      <bmap> ---- bitmap ----
      skyhbt15.pbg
      <rect> ---- click area rect ----
      130
      38
      148
      50
      <curs> ---- cursor ----
      mpfinger.csr
      <help> ---- popup help ----
      Six Button
<endo> ---- end click area ----
<ca13> ---- Seven button ----

```

```

<bgn0> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt16.pbg
    <rect> ---- click area rect ----
    151
    38
    169
    50
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Seven Button
<endo> ---- end click area ----
<ca14> ---- Clear button ----
<bgn0> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt17.pbg
    <rect> ---- click area rect ----
    178
    38
    196
    50
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Clear Button
<endo> ---- end click area ----
<ca15> ---- VFR button ----
<bgn0> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt18.pbg
    <rect> ---- click area rect ----
    205
    38
    223
    50
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    VFR Button
<endo> ---- end click area ----
<ca16> ---- Knob ----
<bgn0> ---- begin click area ----
    <rect> ---- click area rect ----
    184
    4
    225
    36
    <curs> ---- cursor ----
    mpfinger.csr
    <bmap> ---- bitmap ----
    bkxpdrkb.pbg
    <help> ---- popup help ----
    Transponder Mode
<endo> ---- end click area ----
<lite> ---- light flag ----
radi
<help> ---- popup help ----
Bendix/King KT76 Transponder
<endo> ===== END OBJECT =====

```

kt79 King Basic Transponder Gauge

Class: CBasicKingTransponderGauge

Base Class: CBitmapGauge

Description:

Unknown. No examples in any of the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

ky96 King KY-96 Radio Gauge

Class: CBasicKingCommRadioGauge

Base Class: CCommRadioGauge

Description:

Unknown. No examples in any of the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

lbut Lighted Button

Class: CLightButtonGauge

Base Class: CGauge

Description:

This gauge type implements a lighted button. In addition to the normal <mesg> tag (inherited from the 'gage' base class), this gauge type uses the <lmsg> tag to specify a subsystem dependency that controls whether the button is lit or not.

Read Tags

Signature	Description	Type	Format
lmsg	Light Message	OBJECT	Message

Example (TRI200.PNL)

```
<gage> ===== GAUGE ENTRY =====
lbut
<bgn> ===== BEGIN GAUGE ENTRY =====
    <unid> ---- unique id ----
    apfd
    <size> ---- x,y,xsize,ysize ----
    241
    115
    15
    7
    <mesg> ---- switch message ----
    <bgn> ---- begin ----
        <conn> ---- group ----
        K200
        <user> ---- user data ----
        HARDWARE,OTHER
        <user> ---- user data ----
        DATATAG,'fdOn'
    <endo> ---- end ----
    <lmsg> ---- light message ----
    <bgn> ---- begin ----
        <conn> ---- group ----
        K200
        <user> ---- user data ----
        HARDWARE,OTHER
        <user> ---- data tag ----
        DATATAG,'fdOn'
    <endo> ---- end ----
    <bmap> ---- bitmap ----
    pnaapfd.pbg
    0
    0
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    Flight Director
<endo> ===== END GAUGE ENTRY =====
```

lios Lighted In/Out Switch

Class: CLitInOutSwitch

Base Class: CSimpleInOutStateSwitch

Description:

This gauge type implements a lighted version of the 'sios' (Simple In/Out Switch) gauge type. In addition to the normal <mesg> tag (inherited from the 'gage' base class), this gauge type uses the <powr> tag to specify a subsystem dependency that controls whether the button is lit or not.

Read Tags

Signature	Description	Type	Format

Example (TRI220.PNL)

```
<gage> ===== GAUGE ENTRY =====
lios
<bgno> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
    hdg_
  <size> ---- x,y,xsize,ysize ----
    460
    30
    22
    17
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
      AXIS
    <user> ---- user data ----
      HARDWARE,OTHER
    <dtag> ---- user data ----
      _Hdg
  <endo>
  <powr> ---- power message ----
  <bgno> ---- begin ----
    <grou> ---- data tag ----
      K200
    <user> ---- user data ----
      HARDWARE,OTHER
    <user> ---- user data ----
      DATATAG,'actv'
  <endo> ---- end ----
  <out_> ---- out value ----
    0
  <in_> ---- in value ----
    1
  <onsf> ---- on sfx ----
    pushknob.wav
  <ofsfx> ---- off sfx ----
    pushknob.wav
  <sbmp> ---- bitmap name ----
    pnaphdg.pbg
  <crsr> ---- cursor ----
    mpfinger.csr
  <help> ---- popup help ----
    Heading Hold
<endo> ===== END GAUGE ENTRY =====
```



Ilgr Lighted Landing Gear Knob

Class: CLitLandingGearKnobGauge

Base Class: CPushPullKnobGauge

Description:

Read Tags

Signature	Description	Type	Format

Example

lssg Lighted Switch

Class: CLightSwitchSetGauge

Base Class: CGauge

Description:

This gauge type implements a simple lighted switch. In addition to the normal <mesg> tag (inherited from the 'gage' base class), this gauge type uses the <lmsg> tag to specify a subsystem dependency that controls whether the button is lit or not.

Read Tags

Signature	Description	Type	Format
lmsg	Light Message	OBJECT	Message

Example (TRI40.PNL)

```
<gage> ===== GAUGE ENTRY =====
lssg
<bgn> ===== BEGIN GAUGE ENTRY =====
    <unid> ---- unique id ----
    efs1
    <size> ---- x,y,xsize,ysize ----
    556
    647
    23
    11
    <mesg> ---- switch message ----
    <bgn> ---- begin ----
        <conn> ---- group ----
        efs1
        <user> ---- user data ----
        HARDWARE, SWITCH
        <user> ---- user data ----
        UNIT, 1
    <endo> ---- end ----
    <lmsg> ---- light message ----
    <bgn> ---- begin ----
        <conn> ---- group ----
        PEbs
        <user> ---- user data ----
        HARDWARE, BUS
        <dtag> ---- data tag ----
        actv
    <endo> ---- end ----
    <bmap> ---- bitmap ----
    rhes1.pbg
    5
    5
    <curr> ---- cursor ----
    mpfinger.csr
    <help> ---- help ----
    EFIS 1 Power
<endo> ===== END GAUGE ENTRY =====
```

magn Basic Magneto Switch

Class: CBasicMagnetoGauge

Base Class: CGauge

Description:

This gauge type implements a specialized rotary switch for use as a magneto switch.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
magn
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  magn
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- data tag ----
    ignS
    <user> ---- user data ----
    HARDWARE,SWITCH
    <user> ---- user data ----
    ENGINE,1
  <endo>
  <size> ---- switch size ----
  38
  593
  45
  42
  <bmap> ---- bitmap name ----
  skyhmags.pbg
  <curs> ---- right side cursor ----
  mpfinger.csr
  <sfxl> ---- left sfx ----
  s-key.wav
  <sfxr> ---- right sfx ----
  s-key.wav
  <help> ---- popup help ----
  Magneto Switch
<endo> ==== END GAUGE ENTRY ====
```



mchb Piper Malibu CHT Button Gauge

Class: CMalybooCHTButtonGauge

Base Class: CBitmapGauge

Description:

This gauge is a specialized pushbutton designed for the Cylinder Head Temperature select button on the TRI Sahara. There are no gauge-specific data tags defined.

Read Tags

Signature	Description	Type	Format

Example (MALI00.PNL)

```
<gage> ==== GAUGE ENTRY ====
mchb
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    mchb
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- data tag ----
        eMnr
        <user> ---- user data ----
        HARDWARE,OTHER
        <dtag> ---- data tag ----
        fuel
    <endo>
    <size> ---- x,y,xsize,ysize ----
    632
    495
    18
    15
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    CHT Cycle
<endo> ==== END GAUGE ENTRY ====
```



mcht Piper Malibu CHT Indicator Gauge

Class: CMalybooCCHTIndicatorGauge

Base Class: CBitmapGauge

Description:

This gauge is a specialized indicator designed for the Cylinder Head Temperature gauge on the TRI Sahara. This gauge consists of six indicator lights, of which one is illuminated and the rest are unilluminated. The illuminated light indicates which cylinder's temperature is currently being displayed as the CHT value.

Read Tags

Signature	Description	Type	Format
xoff	X pixel offsets	INT	X pixel offset light 1
		INT	X pixel offset light 2
		INT	X pixel offset light 3
		INT	X pixel offset light 4
		INT	X pixel offset light 5
		INT	X pixel offset light 6
yoff	Y pixel offset	INT	Y pixel offset

Example (MALI00.PNL)

```
<gage> ==== GAUGE ENTRY ====
mcht
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  mcht
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
    eMnr
    <user> ---- user data ----
    HARDWARE, OTHER
    <dtag> ---- data tag ----
    fuel
  <endo>
  <size> ---- x,y,xsize,ysize ----
  678
  496
  57
  14
  <bmap> ---- bitmap name ----
  malichtl.pbm
  0
  0
  <xoff> ---- light X offsets ----
  2
  11
  20
  29
  38
  47
  <yoff> ---- light Y offset ----
  4
  <lite> ---- lights ----
  avio
  <help> ---- popup help ----
```



CHT Cylinder
<endo> ==== END GAUGE ENTRY =====

mdcg Piper Malibu Digital Clock Gauge

Class: CMalybooDigitalClockGauge

Base Class: CBitmapGauge

Description:

This is a specialized gauge type for the digital clock in the TRI Sahara.

Read Tags

Signature	Description	Type	Format
mode	Clock Mode	OBJECT	Click Area
powr	Power Select	OBJECT	Click Area
play	Play Button	OBJECT	Click Area
rec_	Record Button	OBJECT	Click Area
set_	Set Clock	OBJECT	Click Area

Example (MALI00.PNL)

```
<gage> ==== GAUGE ENTRY ====
mdcg
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    mdcg
    <size> ---- x,y,xsize,ysize ----
      97
      153
      72
      70
    <clkd> ---- clock display area ----
      20
      12
      56
      21
    <mesg> ---- Subsystem Message ----
    <bgno>
      <conn> ---- data connection ----
      doat
      <user> ---- user data ----
      HARDWARE, GAUGE
      <user> ---- user data ----
      UNIT,1
      <dtag> ---- user data ----
      lmod
    <endo>
    <mode> ---- mode button ----
    <bgno> ---- begin button ----
      <rect> ---- rectangle ----
        21
        2
        48
        10
      <curs> ---- cursor ----
      mpfinger.csr
      <help> ---- popup help ----
      Clock Mode
    <endo> ---- end button
    <powr> ---- power button ----
    <bgno> ---- begin button ----
      <rect> ---- rectangle ----
        20
```



```

53
54
63
< curs> ---- cursor ----
mpfinger.csr
< help> ---- popup help ----
Power Select
< endo> ---- end button
< play> ---- play button ----
< bgno> ---- begin button ----
< rect> ---- rectangle ----
6
31
24
50
< curs> ---- cursor ----
mpfinger.csr
< help> ---- popup help ----
Play Button
< endo> ---- end button
< rec_> ---- rec button ----
< bgno> ---- begin button ----
< rect> ---- rectangle ----
50
31
68
50
< curs> ---- cursor ----
mpfinger.csr
< help> ---- popup help ----
Record Button
< endo> ---- end button
< set_> ---- Set button ----
< bgno> ---- begin button ----
< rect> ---- rectangle ----
24
31
48
50
< curs> ---- cursor ----
mpfinger.csr
< help> ---- popup help ----
Set Clock
< endo> ---- end button
< help> ---- popup help ----
Digital Clock
< mask> ---- light mask ----
pmmask25.raw
< lite> ---- light source ----
avio
< endo> ==== END GAUGE ENTRY ====

```

mcwb Master Caution Button

Class: CMasterCautionButton

Base Class: CGauge

Read Tags

Signature	Description	Type	Format

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
mcwb
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    lmws
  <size> ---- x,y,xsize,ysize ----
    274
    378
    30
    29
  <bmap> ---- bitmap ----
    kamwl.pbg
  <sfx_> ---- switch sfx ----
    switchdn.wav
  <swch> ---- switch message ----
  <bgno> ---- begin ----
    <grou> ---- group ----
      mwsW
      <user> ---- user ----
        HARDWARE, SWITCH
    <endo> ---- end ----
  <lmsg> ---- lite message ----
  <bgno> ---- begin ----
    <grou> ---- group ----
      mwlt
      <user> ---- user ----
        HARDWARE, LIGHT
    <endo> ---- end ----
  <curs> ---- cursor ----
    mpfinger.csr
  <lite> ---- light ----
    anlt
  <help> ---- popup help ----
    Master Warning
<endo> ==== END GAUGE ENTRY ====
```



mfss Piper Malibu Fuel Selector

Class: CMalybooFuelSelectSlider

Base Class: CGauge

Description:

Unknown. No examples in any of the default aircraft. The fuel selector control in the TRI Sahara is implemented as a simple 'swit' type Switch.

Read Tags

Signature	Description	Type	Format

Example

mixt Mixture Knob

Class: CMixtureKnobGauge

Base Class: CPushPullKnobGauge

Description:

This is a specialized version of the 'knob' gauge type, designed for use as a mixture control knob.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ===== GAUGE ENTRY =====
mixt
<bgn> ===== BEGIN GAUGE ENTRY =====
  <unid> ----- unique id -----
  mixt
  <mesg>
  <bgn>
    <conn> ----- data tag -----
    MIXT
    <dtag>
    mixt
  <endo>
  <size> ----- x,y,xsize,ysize -----
  770
  569
  67
  99
  <bmap> ----- bitmap name -----
  skyhmix.pbg
  <poll> ----- polling -----
  <crsr> ----- default cursor -----
  mpslide.csr
  <help> ----- popup help -----
  Engine Mixture
<endo> ===== END GAUGE ENTRY =====
```



mngp Piper Malibu Nav/Gps Panel

Class: CMalybooNavGpsPanelGauge

Base Class: CBitmapGauge

Description:

This gauge type implements the GPS/Nav panel of the TRI Sahara aircraft.

Read Tags

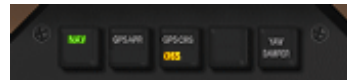
Signature	Description	Type	Format
cnav	NAV Switch	OBJECT	Click Area
xnav	NAV Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
xgps	GPS Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
capr	GPS APR Switch	OBJECT	Click Area
xapr	APR Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
xarm	ARM Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
xact	ACT Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
ccrs	GPS CRS Switch	OBJECT	Click Area
xcrs	CRS Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
xobs	OBS Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
xleg	LEG Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
cmsg	MSG/WPT Switch	OBJECT	Click Area
xmsg	MSG Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
xwpt	WPT Indicator Offset	INT	X pixel offset
		INT	Y pixel offset
cyaw	YAW Switch	OBJECT	Click Area
xyaw	YAW Indicator Offset	INT	X pixel offset
		INT	Y pixel offset

Example (MALI00.PNL)

```

<gage> ==== GAUGE ENTRY ====
mngp
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    mngp
    <size> ---- x,y,xsize,ysize ----
      346
      89
      126
      27
    <cnav> ---- NAV switch ----
    <bgno> ---- begin click area ----
      <rect> ---- click area rect ----
        2
        2

```



```

23
25
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
NAV/GPS Slave
<endo> ---- end click area ----
<xnav> ---- x,y NAV offset ----
7
7
<xgps> ---- x,y GPS offset ----
8
14
<capr> ---- GPS APR switch ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
27
2
50
25
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
GPS Approach
<endo> ---- end click area ----
<xapr> ---- x,y GPS APR offset ----
29
7
<xarm> ---- x,y ARM offset ----
28
15
<xact> ---- x,y ACTV offset ----
35
15
<ccrs> ---- GPS CRS switch ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
51
2
74
25
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
GPS Course
<endo> ---- end click area ----
<xcrs> ---- x,y GPS CRS offset ----
53
7
<xobs> ---- x,y OBS offset ----
54
15
<xleg> ---- x,y LEG offset ----
63
15
<msg> ---- MSG/WPT switch ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
75
2
98
25
<help> ---- popup help ----
GPS Message/Waypoint
<endo> ---- end click area ----
<xmsg> ---- x,y MSG offset ----
81
6
<xwpt> ---- x,y WPT offset ----
81
14

```

```
<cyaw> ---- YAW switch ----
<bgno> ---- begin click area ----
      <rect> ---- click area rect ----
      99
      2
      122
      25
      < curs> ---- cursor ----
      mpfinger.csr
      <help> ---- popup help ----
      Yaw Damper
<endo> ---- end click area ----
<xyaw> ---- x,y YAW offset ----
      102
      8
<mesg> ---- Subsystem Message ----
<bgno>
      <conn> ---- data tag ----
      K150
      <user> ---- user data ----
      HARDWARE, OTHER
<endo>
<help> ---- popup help ----
      NAV/GPS Selector Panel
<endo> ==== END GAUGE ENTRY ====
```

n1ng Turbine N1 Tachometer Dual Needle Gauge

Class: CN1NeedleGauge

Base Class: CTwoNeedleGauge

Description:

This gauge type implements a turbine engine N1 tachometer gauge. It is a derivative of the generic dual-needle gauge '2ned'.

The N1 tachometer value is displayed by the needle specified in the <ndl1> tag, while the <ndl2> tag is used to display a reference N1 value. The reference value can be adjusted by clicking on the knob specified by the <knob> tag.

Read Tags

Signature	Description	Type	Format
knob	Reference Knob	OBJECT	Knob

Example (TRI40.PNL)

```
<gage> ==== GAUGE ENTRY ====
n1ng
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  n1le
  <size> ---- x,y,xsize,ysize ----
  820
  247
  53
  53
  <mesg>
  <bgno>
    <conn>
    L_N1
    <dtag>
    refr
  <endo>
  <ndl1> ---- needle 1 (N1) ----
  <bgno> ---- begin ----
    <cntr> ---- needle center x,y ----
    25
    26
    <mesg>
    <bgno>
      <conn>
      L_N1
      <dtag>
      indn
    <endo>
    <gmin> ---- min value allowed ----
    0.0
    <gmax> ---- max value allowed ----
    120.0
    <nedl> ---- bitmap needle ----
    rhn4
    <bgno> ==== BEGIN BITMAP NEEDLE ====
      <file> ---- needle file ----
      rhneed14.ndl
    <endo> ==== END BITMAP NEEDLE ====
    <sang> ---- starting angle ----
    250.0
```



```

<gtbl> ---- gauge value table ----
<bgn> ---- begin table ----
    <fmt1> - x y
    0.0 0.0
    20.0 37.8
    40.0 74.6
    60.0 110.0
    80.0 145.8
    100.0 182.0
    120.0 219.5
    <endf> - end format
<endo> ---- end table ----
<endo> ---- end ----
<nd12> ---- needle 1 (Bug) ----
<bgn> ---- begin ----
    <cntr> ---- needle center x,y ----
    26
    25
    <mesg>
    <bgn>
        <conn>
        L_N1
        <dtag>
        refr
    <endo>
    <gmin> ---- min value allowed ----
    0.0
    <gmax> ---- max value allowed ----
    120.0
    <ned1> ---- bitmap needle ----
    n1bg
    <bgn> ==== BEGIN BITMAP NEEDLE ====
        <offx> ---- offset X ----
        23
        <offy> ---- offset Y ----
        23
        <base> ---- base bitmap name ----
        rhn1bug
    <endo> ==== END BITMAP NEEDLE ====
    <sang> ---- starting angle ----
    250.0
    <gtbl> ---- gauge value table ----
    <bgn> ---- begin table ----
        <fmt1> - x y
        0.0 0.0
        20.0 37.8
        40.0 74.6
        60.0 110.0
        80.0 145.8
        100.0 182.0
        120.0 219.5
        <endf> - end format
    <endo> ---- end table ----
<endo> ---- end ----
<knob> ---- knob ----
<bgn> ---- begin ----
    <rect> -- rect --
    41
    41
    52
    52
    <bmap> -- bitmap --
    rhn1knob.pbg
    <help> -- help --
    N1 Reference Bug
<endo> ---- end ----
<lite> ---- light ----
    clfl
    <mask> ---- light mask ----
    rhmask11.raw
    <help> ---- help string ----

```

N1 Tach (Left)
<endo> ==== END GAUGE ENTRY =====

namp Piper Navajo Amps Temp Needle Gauge

Class: CJoAmpsTempGauge

Base Class: CNeedleGauge

Description:

This gauge type is a specialized needle gauge intended to display dual ammeter indications for the TRI Kodiak aircraft. This gauge displays the load (amps) on either the left or right alternators.

The user selects which alternator load to display by clicking on either the <ca01> (left) or <ca02> (right) click area buttons. The <mLtS> and <mLtM> tags specify connections to the switch dependency subsystem and the ammeter subsystem for the left alternator. The <mRtS> and <mRtM> tags specify the subsystem connections for the right alternator.

Read Tags

Signature	Description	Type	Format
mChg	Charge Message	OBJECT	Message
mLtM	Left Ammeter Message	OBJECT	Message
mRtM	Right Ammeter Message	OBJECT	Message
mLtS	Left Switch Message	OBJECT	Message
mRtS	Right Switch Message	OBJECT	Message
ca01	Left Switch	OBJECT	Click Area
ca02	Right Switch	OBJECT	Click Area

Example (TRI210.PNL)

```
<gage> ==== GAUGE ENTRY ====
namp
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    amps
  <size> ---- placement and size ----
    470
    146
    89
    100
  <cntr> ---- needle center x,y ----
    45
    64
  <ned1> ---- bitmap needle ----
    stnd
  <bgno> ==== BEGIN BITMAP NEEDLE ====
    <shar> ---- shared needle ----
    <offx> ---- offset X ----
      35
    <offy> ---- offset Y ----
      35
    <base> ---- base bitmap name ----
      NEEDLE1
  <endo> ==== END BITMAP NEEDLE ====
  <gtbl> ---- gauge value table ----
  <bgno> ---- begin table ----
    <fmt1> - x y
      -100.0 0.0
      -50.0 23.5
      0.0 46.0
```



```

50.0 68.0
100.0 91.0
<endf> - end format
<endo> ---- end table ----
<sang> ---- starting angle ----
314.0
<gmin> ---- min value allowed ----
-100.0
<gmax> ---- max value allowed ----
100.0
<mChg> -- Charge Message --
<bgno>
    <conn>
    chrg
    <dtag>
    indn
<endo>
<mLtm> -- Left Ammeter Message --
<bgno>
    <conn>
    ampL
    <dtag>
    indn
<endo>
<mRtm> -- Right Ammeter Message --
<bgno>
    <conn>
    ampR
    <dtag>
    indn
<endo>
<mLts> -- Left Switch Message --
<bgno>
    <conn>
    ampL
    <dtag>
    st8t
<endo>
<mRts> -- Right Switch Message --
<bgno>
    <conn>
    ampR
    <dtag>
    st8t
<endo>
<ca01> ---- left alt button ----
<bgno> ---- begin click area ----
    <rect> ---- click area rect ----
    13
    86
    27
    99
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Left Alternator
<endo> ---- end click area ----
<ca02> ---- right alt button ----
<bgno> ---- begin click area ----
    <rect> ---- click area rect ----
    63
    86
    77
    99
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Right Alternator
<endo> ---- end click area ----
<lite> ---- light ----
plit

```

```
<mask> ---- mask ----  
pnmask29.raw  
<help> ---- popup help ----  
Alternator Load  
<endo> ==== END GAUGE ENTRY ====
```

navg Navigation Gauge

Class: CNavigationGauge

Base Class: CBitmapGauge

Description:

This gauge type implements a standard course deviation indicator (navigation gauge). This gauge displays navigational information from the VOR navigation radio specified in the <radi> tag. This may be VOR or ILS information depending on the frequency tuned and available navigational beacons.

Artwork for the localizer and glideslope needles are specified by the <lndl> and <gndl> tags. The OBS tuning knob is specified with the <obs_> knob object, which rotates the compass card artwork specified by the <comp> tag.

Individual artwork files are specified for the TO, FROM, and FLAG indicator flags via the <ttab>, <ftab> and <wtab> tags respectively. These tags are drawn on the gauge surface at the pixel offset specified by the <ntpl> tag. Similarly, the glideslope flag placement is specified by the <gtpl> tag.

Read Tags

Signature	Description	Type	Format
comp	Compass Plate	ART	Localizer file (PBG)
		INT	X offset
		INT	Y offset
lndl	Localizer needle	ART	Localizer file (PBG)
		INT	X offset
		INT	Y offset
gndl	Glideslope needle	ART	Glideslope art (PBG)
		INT	X offset
		INT	Y offset
obs_	OBS knob	ART	OBS knob art (PBG)
		INT	Top-left X offset
		INT	Top-left Y offset
		INT	Bottom-right X offset
		INT	Bottom-right Y offset
wtab	Warning flag	ART	Warning flag art (PBM)
ttab	'TO' flag	ART	TO flag art (PBM)
ftab	'FROM' flag	ART	FROM flag art (PBM)
ntpl	Nav flag placement	INT	X offset
		INT	Y offset
gtpl	Glideslope flag placement	INT	X offset
		INT	Y offset
radi	Radio info	ID_TYPE	Radio ID
		INT	Unit number
fcs_	FCS message (Use this if gauge is tied to FCS)	OBJECT	Message

Example (FLYHWK00.PNL)

```

<gage> ==== GAUGE ENTRY ====
navg
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    n1ob
  <size> ---- x,y,xsize,ysize ----
    542
    170
    90
    90
  <user> ---- user data ----
    HARDWARE,GAUGE
  <cntr> ---- compass center X,Y ----
    46
    46
  <comp> ---- compass plate needle ----
    navp
  <bgno> ==== BEGIN BITMAP NEEDLE ====
    <file> ---- needle file ----
      csnavcps.ndl
  <endo> ==== END BITMAP NEEDLE ====
  <lndl> ---- localizer needle ----
    cdi1v.pbg
    3
    3
  <gndl> ---- glide slope needle ----
    cdi1h.pbg
    3
    3
  <radi> ---- radio tag ----
    kx15
    1
  <wtab> ---- warning tab ----
    skyhwarn.pbm
  <ttab> ---- TO tab ----
    skyhto.pbm
  <ftab> ---- FR tab ----
    skyhfrom.pbm
  <ntpl> ---- NAV tab placement ----
    40
    62
  <gspl> ---- GS tab placement ----
    55
    38
  <obs_> ---- OBS knob ----
    obsknob.pbg
    -3
    75
    15
    93
    nav1
  <fcs_> ---- FCS switch ----
  <bgno> ---- begin message ----
    <conn> ---- group ----
      K140
    <user> ---- user data ----
      HARDWARE,OTHER
    <dtag> ---- user data ----
      _FCS
  <endo> ---- end message ----
  <crsr> ---- default cursor ----
    mpfinger.csr
  <help> ---- popup help ----
    Course Deviation Indicator
  <lite> ---- light flag ----
    gage
  <mask> ---- light mask ----
    skyhmv1m.raw
<endo> ==== END GAUGE ENTRY ====

```



navr Generic Nav Radio Gauge

Class: CGenericNavRadioGauge

Base Class: CBitmapGauge

Description:

This gauge type implements a generic Nav Radio. The <radi> tag specifies the ID and unit number of the associated radio subsystem.

The active and standby frequencies are each displayed as clickable regions, with each digit being independently settable by clicking left/right on the digit. Click area objects for the active frequency are <ca01> and <ca02> for whole and fractional MHz respectively. Click areas for the standby frequency are <ca03> and <ca04>. The <ca05> (Nav Tune) and <ca06> Nav Transfer button click areas provide an alternative means of changing the frequency; right/left clicking on the Nav Tune click area changes the standby frequency, which can then be toggled with the active frequency by clicking on the Nav Transfer button.

If the <sdim> tag is included in the gauge specification, then the standby frequency is displayed in a dimmed font colour.

Read Tags

Signature	Description	Type	Format
radi	Radio Subsystem	ID_TYPE	NAV Radio Subsystem ID
		INT	Unit number
ca01	Active Whole MHz	OBJECT	Click Area
ca02	Active Fract MHz	OBJECT	Click Area
ca03	Standby Whole MHz	OBJECT	Click Area
ca04	Standby Fract MHz	OBJECT	Click Area
ca05	Nav Tune	OBJECT	Click Area
ca06	Nav Transfer	OBJECT	Click Area
sdim	Standby Dimmed	NO_TYPE	

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
navr
<bgn> ==== BEGIN OBJECT ====
  <unid> ---- unique id ----
  nav1
  <size> ---- x,y,xsize,ysize ----
  690
  710
  69
  71
  <radi> ---- radio identifier ----
  navi
  1
  <sdim> ---- standby digits dimmed ----
  <ca01> ---- NAV active whole digits ----
  <bgn> ---- begin click area ----
    <rect> ---- click area rect ----
    15
    5
    33
    18
```



```

        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        NAV Active Frequency
<endo> ---- end click area ----
<ca02> ---- NAV active fract digits ----
<bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        35
        5
        49
        18
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        NAV Active Frequency
<endo> ---- end click area ----
<ca03> ---- NAV standby whole digits ----
<bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        15
        18
        33
        30
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        NAV Standby Frequency
<endo> ---- end click area ----
<ca04> ---- NAV standby fract digits ----
<bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        35
        18
        49
        30
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        NAV Standby Frequency
<endo> ---- end click area ----
<ca05> ---- NAV tune click area ----
<bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        45
        52
        61
        68
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        NAV Frequency Tune Knob
<endo> ---- end click area ----
<ca06> ---- NAV transfer click area ----
<bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        33
        46
        41
        56
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        NAV Transfer Button
<endo> ---- end click area ----
<help> ---- popup help ----
NAV1 Radio
<endo> ==== END OBJECT ====

```

need Generic Needle Gauge

Class: CNeedleGauge

Base Class: CBitmapGauge

Description:

The 'need' Generic Needle Gauge is a commonly used gauge type which represents an analog indicator needle that rotates based on the gauge value. For example, the RPM indicator gauge on a typical piston-engine aircraft might be implemented using the 'need' gauge type.

Read Tags

Signature	Description	Type	Format
radi	Radius	INT	Pixels
sang	Starting angle	REAL	Degrees (0-360)
dunt	Degrees per unit	REAL	
rang	Angle range (min <= Start Angle <= max)	REAL	Degrees (min)
		REAL	Degrees (max)
nedl	Needle bitmap	ID_TYPE	Needle ID
		OBJECT	Needle
noff	Needle offset	INT	X offset
		INT	Y offset
noac	No auto-center	NO_VALUE	

Example

ngrp NAV Radio Group

Class: CNavRadioGroup

Base Class: CGauge

Description:

Unknown. No examples in any of the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

njfs Piper Navajo Fuel Selector Switch

Class: CJoFuelSelector

Base Class: CBitmapGauge

Description:

This specialized gauge type implements the fuel tank selector switch on the TRI Kodiak aircraft.

This type of fuel selector has three settings : Outboard tank, Off, and Inboard tank. The switch artwork is specified as a three-frame PBG file via the <swit> tag. Three click areas are defined for each of the three switch positions using the <obca>, <ofca> and <ibca> for the Outboard, Off, and Inboard positions respectively. Two subsystem connections are made using the <obsm> and <ibsm> tags. Messages are sent to both subsystems whenever the switch changes state.

Read Tags

Signature	Description	Type	Format
swit	Fuel Switch Bitmap	ART	PBG filename
obca	Outboard Tank Position	OBJECT	Click Area
ofca	Off Position	OBJECT	Click Area
ibca	Inboard Tank Position	OBJECT	Click Area
obsm	Outboard Tank Message	OBJECT	Message
ibsm	Inboard Tank Message	OBJECT	Message

Example (TRI220.PNL)

```
<gage> ==== GAUGE ENTRY ====
njfs
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    nfs1
  <size> ---- x,y,xsize,ysize ----
    408
    619
    91
    82
  <swit> ---- fuel switch bitmap ----
    pnfsell.pbg
  <help> ---- popup help ----
    Fuel Selector Switch (Left)
  <obca> ---- outboard tank click area ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
      0
      0
      30
      82
    <curs> ---- cursor ----
      mpfinger.csr
    <help> ---- popup help ----
      Outboard Tank
  <endo> ---- end click area ----
  <ofca> ---- off tanks click area ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
      31
```



```

0
60
82
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Off
<endo> ---- end click area ----
<ibca> ---- inboard tank click area ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
61
0
91
82
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Inboard Tank
<endo> ---- end click area ----
<obms> ---- outboard message ----
<bgno> ---- begin message ----
<grou> ---- group ID ----
Ltko
<dtag> ---- Data Tag ----
st8t
<endo> ---- end message ----
<ibms> ---- inboard message ----
<bgno> ---- begin message ----
<grou> ---- group ID ----
Ltki
<dtag> ---- Data Tag ----
st8t
<endo> ---- end message ----
<endo> ==== END GAUGE ENTRY ====

```

nral Piper Navajo Radar Altimeter Needle Gauge

Class: CJoRadarAltimeterGauge

Base Class: CNeedleGauge

Description:

This gauge type implements a radar altimeter gauge. The actual radiometric altitude is displayed using the <nedl> tag inherited from the base 'need' gauge type. Specialized tags for this gauge type are used to control the decision height alarm and indication.

Read Tags

Signature	Description	Type	Format
bMsg	DH Bug Message	OBJECT	Message
dhbt	DH Bug Table	OBJECT	Table
bug	DH Bug	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
plit	Power Light	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
dhms	DH Message	OBJECT	Message
knob	DH Knob	OBJECT	Knob

Example (TRI200.PNL)

```
<gage> ==== GAUGE ENTRY ====
nral
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  rAlt
  <size> ---- x,y,xsize,ysize ----
  342
  312
  90
  90
  <mesg> -- Connect to Radar Altimeter Subsystem --
  <bgno>
    <conn>
    alt3
    <dtag>
    rAlt
  <endo>
  <bMsg> -- Connect to Radar Altimeter Decision Height Bug Subsystem --
  <bgno>
    <conn>
    alt3
    <dtag>
    dh__
  <endo>
  <cntr> ---- needle center x,y ----
  44
  46
  <gmin> ---- min value allowed ----
  0.0
  <gmax> ---- max value allowed ----
  2500.0
  <rang> ---- angle range ----
  0.0
```



```

247.0
<nedl> ---- bitmap needle ----
nrda
<bgno> ==== BEGIN BITMAP NEEDLE ====
    <shar> ---- shared needle ----
    <base> ---- base bitmap name ----
    navjrand
<endo> ==== END BITMAP NEEDLE ====
<gtbl> ---- gauge value table ----
<bgno> ---- begin table ----
    <fmt1> - x y
    0.0 0.0
    100.0 26.8
    200.0 63.4
    300.0 96.2
    400.0 130.3
    500.0 169.0
    1000.0 188.0
    1500.0 208.0
    2000.0 228.0
    2500.0 247.0
    <endf> - end format
<endo> ---- end table ----
<dhbt> ---- decision height bug table ----
<bgno> ---- begin table ----
    <fmt1> - x y
    0.0 0.0
    26.8 100.0
    63.4 200.0
    96.2 300.0 96.2
    130.3 400.0
    169.0 500.0
    188.0 1000.0
    208.0 1500.0
    228.0 2000.0
    247.0 2500.0
    <endf> - end format
<endo> ---- end table ----
<bug > ---- decision height bug ----
navjrabg.pbg
3
3
<plit> ---- power light ----
navjralt.pbg
24
16
<dhms> ---- decision height message ----
<bgno> ---- begin message ----
    <conn> ---- group ID ----
    'dehl'
    <user> ---- user data ----
    HARDWARE,LIGHT
<endo> ---- end message ----
<knob> ---- true airspeed knob ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    navjrakb.pbg
    <rect> ---- click area rect ----
    2
    73
    17
    88
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Decision Height Bug
<endo> ---- end click area ----
<dfmt> ---- display format ----
%.0f
<unit> ---- display units ----
FEET

```

```
<lite> ---- light ----  
plit  
<mask> ---- mask ----  
pnmask26.raw  
<help> ---- popup help ----  
Radar Altimeter  
<endo> ==== END GAUGE ENTRY ====
```

nrdg Nav Radio Gauge

Class: CNavRadioGauge

Base Class: CBitmapGauge

Description:

Unknown. No examples in any of the default aircraft.

Read Tags

Signature	Description	Type	Format

Example

ogrp NAV Gauge Group

Class: CNavGaugeGroup

Base Class: CGauge

Description:
Unknown?

Read Tags

Signature	Description	Type	Format
nav1	NAV 1 OBS ID	ID_TYPE	
nav2	NAV 2 OBS ID	ID_TYPE	

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====  
ogrp  
<bgn> ==== BEGIN GAUGE ENTRY ====  
    <nav1> ---- nav 1 OBS identifier ----  
        n1ob  
    <nav2> ---- nav 2 OBS identifier ----  
        n2ob  
<endo> ==== END GAUGE ENTRY =====
```

oilp Oil Pressure Needle Gauge

Class: COilPressureGauge

Base Class: CNeedleGauge

Description:

This gauge type is intended to implement an oil pressure needle gauge. There are no gauge-specific tags.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
oilp
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    oilp
  <size> ---- x,y,xsize,ysize ----
    63
    375
    33
    66
  <cntr> ---- needle center x,y ----
    18
    32
  <over> ---- overlay image ----
    skyhoilp.pbm
    0
    0
  <user> ---- user data ----
    HARDWARE,GAUGE
  <user> ---- user data ----
    ENGINE,1
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- connected to ----
      oilP
  <endo>
  <nedl> ---- bitmap needle ----
    smal
  <bgno> ==== BEGIN BITMAP NEEDLE ====
    <file> ---- needle file ----
      needle3.ndl
  <endo> ==== END BITMAP NEEDLE ====
  <gtbl> ---- gauge value table ----
  <bgno> ---- begin table ----
    <fmt1> - x y
      0.0 197.4
      20.0 222.0
      40.0 249.9
      60.0 277.3
      80.0 305.5
      100.0 325.9
      115.0 343.8
    <endf> - end format
  <endo> ---- end table ----
  <gmin> ---- min value allowed ----
    0.0
  <gmax> ---- max value allowed ----
```



```
115.0
<help> ---- popup help ----
Oil Pressure
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhoprm.raw
<endo> ==== END GAUGE ENTRY ====
```

oilt Oil Temperature Needle Gauge

Class: COilTemperatureGauge

Base Class: CNeedleGauge

Description:

This gauge type is intended to implement an oil temperature needle gauge. There are no gauge-specific tags.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ===== GAUGE ENTRY =====
oilt
<bgno> ===== BEGIN GAUGE ENTRY =====
    <unid> ---- unique id ----
        oilt
    <size> ---- x,y,xsize,ysize ----
        30
        375
        33
        66
    <cntr> ---- needle center x,y ----
        14
        32
    <over> ---- overlay image ----
        skyhoilt.pbm
        0
        0
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- connected to ----
            oilt
        <user> ---- user data ----
            HARDWARE, GAUGE
        <user> ---- user data ----
            ENGINE,1
    <endo>
    <nedl> ---- bitmap needle ----
        smal
    <bgno> ===== BEGIN BITMAP NEEDLE =====
        <file> ---- needle file ----
            needle3.ndl
    <endo> ===== END BITMAP NEEDLE =====
    <gtbl> ---- gauge value table ----
    <bgno> ---- begin table ----
        <fmt1> - x y
            0.0 163.1
            100.0 150.7
            150.0 107.2
            200.0 58.2
            240.0 18.4
        <endf> - end format
    <endo> ---- end table ----
    <gmin> ---- min value allowed ----
        0.0
    <gmax> ---- max value allowed ----
        240.0
    <help> ---- popup help ----
```



```
Oil Temperature
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhotpm.raw
<endo> ==== END GAUGE ENTRY ====
```

pair Switch Pair

Class: CSwitchPairGauge

Base Class: CSimpleSwitchGauge

Description:

This switch type implements a pair of independent switches within one gauge object. Each switch has its own click areas defined to turn the switch off or on (<s1of> and <s1on>, <s2of> and <s2on>) as well as independent subsystem messages (<s1ms> and <s2ms>).

Read Tags

Signature	Description	Type	Format
s1ms	Switch #1 Message	OBJECT	Message
s2ms	Switch #2 Message	OBJECT	Message
s1of	Switch #1 Off	OBJECT	Click Area
s1on	Switch #1 On	OBJECT	Click Area
s2of	Switch #2 Off	OBJECT	Click Area
s2on	Switch #2 On	OBJECT	Click Area

Example (TRI226.PNL)

```
<gage> ==== GAUGE ENTRY ====
pair
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  altL
  <s1ms> ---- switch #1 message ----
  <bgno> ---- begin ----
    <conn> ---- group ----
    altL
    <user> ---- hardware ----
    HARDWARE, SWITCH
  <endo> ---- end ----
  <s2ms> ---- switch #2 message ----
  <bgno> ---- begin ----
    <conn> ---- group ----
    altR
    <user> ---- hardware ----
    HARDWARE, SWITCH
  <endo> ---- end ----
  <size> ---- x,y,xsize,ysize ----
  134
  578
  140
  103
  <s1of> ---- switch #1 off area ----
  <bgno> ---- begin ----
    <rect> ---- rect ----
    0
    0
    67
    48
    <curs> ---- cursor ----
    mpfliplt.csr
    <help> ---- text ----
    Off (Left)
  <endo> ---- end ----
  <s1on> ---- switch #1 on area ----
  <bgno> ---- begin ----
```



```

        <rect> ---- rect ----
        68
        0
        139
        48
        <curs> ---- cursor ----
        mpfliprt.csr
        <help> ---- text ----
        On (Left)
    <endo> ---- end ----
    <s2of> ---- switch #2 off area ----
    <bgn> ---- begin ----
        <rect> ---- rect ----
        0
        49
        67
        102
        <curs> ---- cursor ----
        mpfliplr.csr
        <help> ---- text ----
        Off (Right)
    <endo> ---- end ----
    <s2on> ---- switch #2 on area ----
    <bgn> ---- begin ----
        <rect> ---- rect ----
        68
        49
        139
        102
        <curs> ---- cursor ----
        mpfliprt.csr
        <help> ---- text ----
        On (Right)
    <endo> ---- end ----
    <bmap> ---- bitmap name ----
    navjalcb.pbg
    <sfxu> ---- up sfx ----
    switchup.wav
    <sfxd> ---- down sfx ----
    switchdn.wav
    <sstr> ---- switch strings ----
    4
    Left (Off)/Right (Off)
    Left (On)/Right (Off)
    Left (Off)/Right (On)
    Left (On)/Right (On)
    <help> ---- popup help ----
    Alternator
    <endo> ==== END GAUGE ENTRY ====

```

pefs Pilatus EFIS Panel

Class: CPilatusEFISPanel

Base Class: CGauge

Description:

This customized gauge implements the EFIS control panel for the Pilatus PC-12 avionics suite.

Read Tags

Signature	Description	Type	Format
pfdm	PFD Message	OBJECT	Message
navm	ND Message	OBJECT	Message
dh__	DH Knob	OBJECT	Knob
brt_	Brightness Knob	OBJECT	Knob
crs_	Course Knob	OBJECT	Knob
hdg_	Heading Knob	OBJECT	Knob
hsi_	HSI Button	OBJECT	Knob
arc_	ARC Button	OBJECT	Knob
nav_	NAV Button	OBJECT	Knob
rgup	Range Up Button	OBJECT	Knob
rgdn	Range Down Button	OBJECT	Knob
nav1	NAV1 Button	OBJECT	Knob
nav2	NAV2 Button	OBJECT	Knob
1-2_	1-2 System Select Button	OBJECT	Knob

Example (PIL_MAIN.PNL)

```
<gage> ===== GAUGE ENTRY =====
pefs
<bgno> ===== Pilatus EFIS Panel =====
  <unid> ----- unique id -----
  pefs
  <size> ----- x,y,xsize,ysize -----
  721
  677
  162
  50
  <pfdm> ----- Primary Flight Display Message -----
  <bgno>
    <conn>
    ppfd
    <dtag>
    st8t
  <endo>
  <navm> ----- Nav Display Message -----
  <bgno>
    <conn>
    pnvd
    <dtag>
    st8t
  <endo>
  <dh__> ----- Decision Height Knob -----
  <bgno> ----- begin click area -----
    <rect> ----- click rect -----
    15
    2
    29
```



```

18
<bmap> ---- bitmap ----
pefspkna.pbg
< curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Decision Height Knob
<endo> ---- end click area ----
<brt_> ---- Brightness Knob ----
<bgno> ---- begin click area ----
<rect> ---- click rect ----
133
1
147
17
<bmap> ---- bitmap ----
pefspknb.pbg
< curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Display Brightness Knob
<endo> ---- end click area ----
<crs_> ---- Course Knob ----
<bgno> ---- begin click area ----
<rect> ---- click rect ----
12
29
30
48
<bmap> ---- bitmap ----
pefspknc.pbg
< curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Course Select Knob
<endo> ---- end click area ----
<hdg_> ---- Heading Knob ----
<bgno> ---- begin click area ----
<rect> ---- click rect ----
131
29
149
48
<bmap> ---- bitmap ----
pefspknd.pbg
< curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Heading Select Knob
<endo> ---- end click area ----
<hsi_> ---- HSI Button ----
<bgno> ---- begin click area ----
<rect> ---- click rect ----
37
12
57
28
<bmap> ---- bitmap ----
pefsphsi.pbg
< curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
HSI 360deg mode selector
<endo> ---- end click area ----
<arc_> ---- ARC Button ----
<bgno> ---- begin click area ----
<rect> ---- click rect ----
59
12
79
28

```

```

        <bmap> ---- bitmap ----
        pefsparc.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Arc mode selector
<endo> ---- end click area ----
<nav_> ---- NAV Button ----
<bgno> ---- begin click area ----
        <rect> ---- click rect ----
        82
        12
        102
        28
        <bmap> ---- bitmap ----
        pefspnav.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        NAV Sensor Selector
<endo> ---- end click area ----
<rgup> ---- Range Up Button ----
<bgno> ---- begin click area ----
        <rect> ---- click rect ----
        104
        12
        124
        28
        <bmap> ---- bitmap ----
        pefsprup.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Range Selection -> Up
<endo> ---- end click area ----
<rgdn> ---- Range Down Button ----
<bgno> ---- begin click area ----
        <rect> ---- click rect ----
        104
        31
        124
        47
        <bmap> ---- bitmap ----
        pefsprdn.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Range Selection -> Down
<endo> ---- end click area ----
<nav1> ---- Nav 1 Button ----
<bgno> ---- begin click area ----
        <rect> ---- click rect ----
        37
        31
        57
        47
        <bmap> ---- bitmap ----
        pefspar1.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        No 1 System Bearing Pointer Select
<endo> ---- end click area ----
<nav2> ---- Nav 2 Button ----
<bgno> ---- begin click area ----
        <rect> ---- click rect ----
        60
        31
        80
        47
        <bmap> ---- bitmap ----

```

```

        pefspar2.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        No 2 System Bearing Pointer Select
<endo> ---- end click area ----
<1-2_> ---- 1-2 System Select Button ----
<bgn> ---- begin click area ----
        <rect> ---- click rect ----
        82
        31
        102
        47
        <bmap> ---- bitmap ----
        pefsp1-2.pbg
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        1-2 System Select
<endo> ---- end click area ----
<lite> ---- light flag ----
silt
<help> ---- popup help ----
Pilot EFIS Controls
<endo> ==== END OBJECT ====

```

pmem Piper Malibu Engine Monitor

Class: CMalybooEngineMonitorGauge

Base Class: CGauge

Description:

This gauge type implements the engine monitoring panel in the TRI Sahara aircraft.

Read Tags

Signature	Description	Type	Format
fuel	Fuel Button	OBJECT	Click Area
inst	Inst Button	OBJECT	Click Area
blk_	Blk Button	OBJECT	Click Area
fill	Fill Button	OBJECT	Click Area
pwr_	Pwr% Button	OBJECT	Click Area
set_	SET Button	OBJECT	Click Area
map_	Manifold Pressure	OBJECT	Click Area
RPM_	Engine RPM	OBJECT	Click Area
TIT	Intake Temperature	OBJECT	Click Area
FF_	Fuel Flow	OBJECT	Click Area
OT_	Oil Temperature	OBJECT	Click Area
OP_	Oil Pressure	OBJECT	Click Area
CHT_	Cylinder Head Temp	OBJECT	Click Area
VAC_	Vacuum Pressure	OBJECT	Click Area
LQTY	Left Fuel Quantity	OBJECT	Click Area
RQTY	Right Fuel Quantity	OBJECT	Click Area
rlit	Red Light Artwork	ART	PBM filename
glit	Green Light Artwork	ART	PBM filename
ylit	Yellow Light Artwork	ART	PBM filename

Example (MALI00.PNL)

```

<gage> ==== GAUGE ENTRY ====
pmem
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    pmem
    <mesg> ---- Subsystem Message ----
    <bgno>
      <conn> ---- data tag ----
      eMnr
      <user> ---- user data ----
      HARDWARE, OTHER
      <dtag> ---- data tag ----
      fuel
    <endo>
    <size> ---- x,y,xsize,y size ----
    631
    115
    135
    369
    <fuel> ---- Fuel button ----
    <bgno> ---- begin click ----
      <rect> ---- rect ----
      3
      20
      9

```



```

31
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
Fuel button
<endo> ---- end click ----
<inst> ---- Inst button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
3
34
9
45
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
Inst button
<endo> ---- end click ----
<blk_> ---- Blk button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
3
48
9
59
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
Blk button
<endo> ---- end click ----
<fill> ---- Fill button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
126
20
132
31
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
Fill button
<endo> ---- end click ----
<pwr_> ---- Pwr% button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
126
34
132
45
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
Pwr% button
<endo> ---- end click ----
<set_> ---- Set button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
126
48
132
59
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
Set button
<endo> ---- end click ----
<MAP_> ---- MAP button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
1
104

```

```

7
115
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
MAP button
<endo> ---- end click ----
<RPM_> ---- RPM button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
128
104
134
115
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
RPM button
<endo> ---- end click ----
<TIT_> ---- TIT button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
1
162
7
173
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
TIT button
<endo> ---- end click ----
<FF_> ---- FF button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
128
162
134
173
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
FF button
<endo> ---- end click ----
<OT_> ---- OT button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
1
219
7
230
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
OT button
<endo> ---- end click ----
<OP_> ---- OP button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
128
219
134
230
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
OP button
<endo> ---- end click ----
<CHT_> ---- CHT button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
1

```

```

276
7
287
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
CHT button
<endo> ---- end click ----
<VAC_> ---- VAC button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
128
276
134
287
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
VAC button
<endo> ---- end click ----
<LQTY> ---- LQTY button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
1
334
7
345
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
LQTY button
<endo> ---- end click ----
<RQTY> ---- RQTY button ----
<bgno> ---- begin click ----
<rect> ---- rect -----
128
334
134
345
<curs> --- cursor ---
mpfinger.csr
<help> ---- help ----
RQTY button
<endo> ---- end click ----
<rlit> ---- red light ----
pmsrlite.pbm
<glit> ---- green light ----
pmsglite.pbm
<ylit> ---- yellow light ----
pmsylite.pbm
<help> ---- popup help ----
Engine Monitor/Digital Readout
<endo> ==== END GAUGE ENTRY ====

```

prim Primer Knob

Class: CPrimerKnobGauge

Base Class: CPushPullKnobGauge

Description:

This gauge type implements a specialized type of push/pull knob, intended for use as a fuel primer knob. There are no gauge-specific tags.

Read Tags

Signature	Description	Type	Format

Example (STRMAIN.PNL)

```
<gage> ==== GAUGE ENTRY ====
prim
<bgn> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    prim
    <size> ---- x,y,xsize,ysize ----
    410
    927
    72
    106
    <mesg>
    <bgn>
        <conn> ---- Primer Control ----
        prim
        <dtag>
        prim
    <endo>
    <bmap> ---- bitmap name ----
    str_prmr.pbg
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Primer
<endo> ==== END GAUGE ENTRY ====
```



push Pushbutton Switch

Class: CPushButtonGauge

Base Class: CGauge

Description:

This common gauge type implements a simple push-button. The switch state is Active (On) when the button is pressed, and Inactive (Off) when the button is released. The <mmnt> tag may be to specify that the button is momentary-contact, such that the switch is only in the On state while the user is actively pressing on the button by clicking the mouse. If the <mmnt> tag is not specified, then the button stays locked in the new state every time it is pressed.

Read Tags

Signature	Description	Type	Format
mmnt	Momentary Contact	NO_TYPE	
bmap	Artwork	ART	PBG filename
curs	Cursor	CURSOR	CSR filename

Example (TRI42.PNL)

```
<gage> ==== GAUGE ENTRY ====
push
<bgn> ==== BEGIN GAUGE ENTRY ====
  <size> ---- x,y,xsize,ysize ----
    400
    143
    19
    28
  <dtyp> -- Default Type --
    INT
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- connected to ----
    si3T
    <user> ---- user data ----
    HARDWARE,SWITCH
  <endo>
  <bmap> ---- bitmap ----
    rhovpt2.pbg
  <mmnt> ---- momentary ----
  <curs> ---- cursor ----
    mpfinger.csr
  <help> ---- popup help ----
    Stall Ident 3 Test
<endo> ==== END GAUGE ENTRY ====
```



radg Radio Gauge Group

Class: CRadioGaugeGroup

Base Class: CGauge

Description:
Unknown?

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====  
radg  
<bgn> ==== BEGIN GAUGE ====  
    <rad1> ---- radio #1 ----  
    nc01  
    <rad2> ---- radio #2 ----  
    nc02  
    <key_> ---- key ----  
    acom  
<endo> ==== END GAUGE =====
```

revt Reversible Throttle Knob

Class: CReversibleThrottleKnobGauge

Base Class: CPushPullKnobGauge

Description:

This specialized push/pull knob is intended for use as a throttle control which supports thrust reversers integrated into the throttle.

Read Tags

Signature	Description	Type	Format
rvsr	Reverser	OBJECT	Message
rbmp	Reverse Bitmaps	ART	PBG filename
ytop	Top Y Pixel	INT	Y pixel
ybtm	Bottom Y Pixel	INT	Y pixel
rtop	Top Y Pixel - Reverse	INT	Y pixel
rbtm	Bottom Y Pixel - Reverse	INT	Y pixel

Example (TRI41.PNL)

```
<gage> ===== GAUGE ENTRY =====
revt
<bgno> ===== BEGIN GAUGE ENTRY =====
  <unid> ---- unique id ----
  thr1
  <mesg> ---- message data ----
  <bgno> ---- begin ----
    <conn>
    Thr1
    <dtag>
    thro
  <endo> ---- end ----
  <lnk1> ---- linked message ----
  <bgno> ---- begin ----
    <conn>
    Thr2
    <dtag>
    thro
  <endo> ---- end ----
  <rvsr> ---- reverser ----
  <bgno> ---- begin ----
    <conn> ---- group ----
    Rev1
  <endo> ---- end ----
  <size> ---- x,y,xsize,ysize ----
  450
  48
  69
  263
  <bmap> ---- bitmap name ----
  rhthrol.pbg
  <rbmp> ---- reverse bitmaps ----
  rhthrolr.pbg
  <poll> ---- polling ----
  <ytop> ---- top y click ----
  44
  <ybtm> ---- bottom y click ----
  263
  <rtop> ---- top y click (reverse) ----
  90
```



```
<rbtn> ---- bottom y click (reverse) ----  
183  
<crsr> ---- default cursor ----  
mpslide.csr  
<help> ---- popup help ----  
Engine Throttle (Left)  
<endo> ==== END GAUGE ENTRY =====
```

rhfp Raytheon Hawker Fire System

Class: C???

Base Class: CGauge

Description:

This gauge implements the fire control system panel on the TRI Peregrine 800TR jet.

Read Tags

Signature	Description	Type	Format
fir1	Engine 1 Fire	OBJECT	Message
fir2	Engine 2 Fire	OBJECT	Message
ext1	Extinguisher 1	OBJECT	Knob
ext2	Extinguisher 2	OBJECT	Knob
bel1	Bell 1 Cancel	OBJECT	Click Area
bel2	Bell 2 Cancel	OBJECT	Click Area
ef1_	Ext 1 Fired Light	OBJECT	Rectangle
ef2_	Ext 2 Fired Light	OBJECT	Rectangle
ann1	Fire Annunciator 1	OBJECT	Annunciator
ann2	Fire Annunciator 2	OBJECT	Annunciator

Example (TRI42.PNL)

```
<gage> ==== GAUGE ENTRY ====
rhfp
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  rhfp
  <fir1> ---- data provider ----
  <bgno> ---- begin ----
    <conn> ---- group ----
    Fir1
    <user>
    HARDWARE,OTHER
    <dtag> ---- data tag ----
    fire
  <endo> ---- end ----
  <fir2> ---- data provider ----
  <bgno> ---- begin ----
    <conn> ---- group ----
    Fir2
    <user>
    HARDWARE,OTHER
    <dtag> ---- data tag ----
    fire
  <endo> ---- end ----
  <size> ---- x,y,xsize,ysize ----
  484
  531
  212
  75
  <ext1> ---- Eng 1 Ext ----
  <bgno> ---- begin ----
    <rect>
    16
    12
    44
    67
    <bmap>
```



```

        rhove1e.pbg
        < curs>
        mpfinger.csr
        < help>
        Engine 1 Extinguish
<endo> ---- end ----
<ext2> ---- Eng 2 Ext ----
<bgn> ---- begin ----
        < rect>
        166
        12
        194
        67
        < bmap>
        rhove2e.pbg
        < curs>
        mpfinger.csr
        < help>
        Engine 2 Extinguish
<endo> ---- end ----
<bel1> ---- Bell Cancel 1 ----
<bgn> ---- begin ----
        < rect>
        82
        43
        103
        63
        < curs>
        mpfinger.csr
        < help>
        Bell Cancel
<endo> ---- end ----
<bel2> ---- Bell Cancel 2 ----
<bgn> ---- begin ----
        < rect>
        108
        43
        129
        63
        < curs>
        mpfinger.csr
        < help>
        Bell Cancel
<endo> ---- end ----
<ef1_> ---- Fired light 1 ----
<bgn> ---- begin ----
        < rect>
        55
        45
        68
        58
<endo> ---- end ----
<ef2_> ---- Fired light 2 ----
<bgn> ---- begin ----
        < rect>
        143
        45
        156
        58
<endo> ---- end ----
<ann1> ---- Fire Annunc 1 ----
<bgn> ---- begin ----
        < rect>
        58
        13
        81
        23
        < bmap>
        rhe1fire.pbg
<endo> ---- end ----
<ann2> ---- Fire Annunc 2 ----

```

```
<bgno> ---- begin ----
      <rect>
      130
      13
      153
      23
      <bmap>
      rhe2fire.pbg
<endo> ---- end ----
<help> ---- help ----
      Fire Extinguish Panel
<endo> ==== END GAUGE ENTRY ====
```

rock Rocker Switch

Class: CRockerSwitchGauge

Base Class: CGauge

Description:

This gauge type implements a rocker switch. This switch has a normal centered position, plus two rocker states which have independent messaging subsystems.

The first (Up) state click area is defined by the <one_> tag, which uses the <msg1> message to inform the appropriate subsystem. Similarly, the second (Down) state click area is defined by the <two_> tag and <msg2> indicates the subsystem messaging target.

Read Tags

Signature	Description	Type	Format
one_	First Click Area	OBJECT	Click Area
two_	Second Click Area	OBJECT	Click Area
msg1	Message One	OBJECT	Message
msg2	Message Two	OBJECT	Message

Example (TRI220.PNL)

```
<gage> ==== GAUGE ENTRY ====
rock
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  updn
  <size> ---- x,y,xsize,ysize ----
  429
  32
  21
  36
  <one_> ---- DN click area ----
  <bgno> ---- begin ----
    <rect> ---- rect ----
    0
    18
    21
    36
    <curs> ---- cursor ----
    mpflipdn.csr
  <endo> ---- end ----
  <two_> ---- UP click area ----
  <bgno> ---- begin ----
    <rect> ---- rect ----
    0
    0
    21
    17
    <curs> ---- cursor ----
    mpflipup.csr
  <endo> ---- end ----
  <msg2> ---- message 2 ----
  <bgno> ---- begin ----
    <grou> ---- data tag ----
    AXIS
    <user> ---- user data ----
    HARDWARE, OTHER
```



```
        <user> ---- user data ----
        DATATAG, '__dn'
<endo> ---- end ----
<msg1> ---- message 1 ----
<bgno> ---- begin ----
        <grou> ---- data tag ----
        AXIS
        <user> ---- user data ----
        HARDWARE, OTHER
        <user> ---- user data ----
        DATATAG, '__up'
<endo> ---- end ----
<bmap> ---- bitmap name ----
pnaprte.pbg
<help> ---- popup help ----
Climb Rate
<endo> ==== END GAUGE ENTRY ====
```

rmig Remote Magnetic Indicator Gauge

Class: CRMINeedleGauge

Base Class: CBitmapGauge

Description:

The RMI gauge displays both ADF and VOR directional information on a single gauge.

The <mesg> tag specifies the subsystem that supplies the current aircraft magnetic heading. This is used to display the underlying compass card for the gauge. The artwork for the compass card is specified with the <comp> tag, and its offset within the gauge surface is specified with the <cpy> tag.

Read Tags

Signature	Description	Type	Format
comp	Compass Plate	ID_TYPE	
		OBJECT	Needle
cpy	Compass plate (X, Y)	INT	X offset
		INT	Y offset
adfn	ADF needle	OBJECT	Needle
anxy	ADF needle (X, Y)	INT	X offset
		INT	Y offset
rmin	RMI needle	OBJECT	Needle
rnxy	RMI needle (X, Y)	INT	X offset
		INT	Y offset
adfr	ADF radio	ID_TYPE	ADF radio
adf1		INT	unit number
adf2	2 nd ADF radio	ID_TYPE	ADF radio
		INT	unit number
navr	1 st NAV radio	ID_TYPE	NAV radio
nav1		INT	Unit number
nav2	2 nd NAV radio	ID_TYPE	NAV radio
		INT	Unit number
adfa	ADF arrow	ART	ADF Arrow (PBG)
		INT	X offset
		INT	Y offset
vora	VOR arrow	ART	VOR Arrow (PBG)
		INT	X offset
		INT	Y offset
lbtn	Left button	OBJECT	Click Area
rbtn	Right button	OBJECT	Click Area

Example (TRI200.PNL)

```
<gage> ==== GAUGE ENTRY ====
rmig
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    rmig
    <mesg>
```



```

<bgno>
    <conn>
    hsi1
    <dtag>
    yaw_
<endo>
<size> ---- x,y,xsize,ysize ----
18
312
90
90
<over> ---- overlay image ----
navjadfo.pbm
-6
-2
<comp> ---- compass plate ----
gyro
<bgno> ==== BEGIN NEEDLE ====
    <offx> ---- offset X ----
    43
    <offy> ---- offset Y ----
    43
    <base> ---- base bitmap name ----
    DGYR0
<endo> ==== END NEEDLE ====
<adfn> ---- ADF needle ----
adfn
<bgno> ==== BEGIN NEEDLE ====
    <offx> ---- offset X ----
    30
    <offy> ---- offset Y ----
    30
    <base> ---- base bitmap name ----
    adfneed
<endo> ==== END NEEDLE ====
<adfr> ---- ADF radio ----
adfr
1
<rmin> ---- RMI (nav) needle ----
rmin
<bgno> ==== BEGIN NEEDLE ====
    <offx> ---- offset X ----
    43
    <offy> ---- offset Y ----
    43
    <base> ---- base bitmap name ----
    needlrmi
<endo> ==== END NEEDLE ====
<nav2> ---- NAV radio ----
kx15
2
<lite> ---- light ----
plit
<mask> ---- mask ----
pnmask12.raw
<help> ---- popup help ----
Auto Direction Finder
<endo> ==== END GAUGE ENTRY ====

```

selt Flyhawk Emergency Locator Transmitter Switch

Class: CFlyhawkELTSwitch

Base Class: CBitmapGauge

Description:

This gauge type implements a three-state Emergency Location Transmitter switch. The three states are "On", "Auto" and "Reset".

Read Tags

Signature	Description	Type	Format
ulit	Unlit Artwork	ART	PBG filename
lit	Lit Artwork	ART	PBG filename
csru	Up Cursor	CURSOR	CSR filename
csrd	Down Cursor	CURSOR	CSR filename
sfxu	Up Sound Effect	SOUND	WAV filename
sfxd	Down Sound Effect	SOUND	WAV filename

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
selt
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  selt
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
    elts
    <user> ---- user data ----
    HARDWARE, SWITCH
  <endo>
  <size> ---- x,y,xsize,ysize ----
  1238
  188
  38
  37
  <ulit> ---- bitmap name ----
  skyhelt0.pbg
  <lit> ---- bitmap name ----
  skyhelt1.pbg
  <csru> ---- up cursor ----
  mpflipup.csr
  <csrd> ---- down cursor ----
  mpflipdn.csr
  <sfxu> ---- up sfx ----
  switchup.wav
  <sfxd> ---- down sfx ----
  switchdn.wav
  <help> ---- popup help ----
  Emergency Locator
<endo> ==== END GAUGE ENTRY =====
```



shan Skyhawk Annunciator Panel

Class: CSkyhawkAnnunciatorPanel

Base Class: CBitmapGauge

Description:

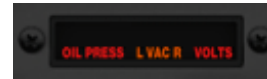
This specific indicator panel gauge implements the caution/warning annunciator panel for the TRI Skyhawk. It is specifically hard-coded to use "known" subsystems for this aircraft, so caution should be taken when using it in other aircraft that all annunciator subsystems be re-used as well.

Read Tags

Signature	Description	Type	Format
lowf	Low Fuel	ART	PBM filename
lowl	Low Fuel Left	ART	PBM filename
lowr	Low Fuel Right	ART	PBM filename
oilp	Oil Pressure	ART	PBM filename
vacl	Low Vacuum Left	ART	PBM filename
vacr	Low Vacuum Right	ART	PBM filename
lowv	Low Vacuum	ART	PBM filename
volt	System Volts	ART	PBM filename
line	Warning Line	ART	PBM filename

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
shan
<bgn> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    anuc
    <size> ---- x,y,xsize,ysize ----
    386
    131
    97
    27
    <lowf> ---- low fuel ----
    skyhlowf.pbm
    <lowl> ---- low fuel L ----
    skyhlowl.pbm
    <lowr> ---- low fuel R ----
    skyhlowr.pbm
    <oilp> ---- oil pressure ----
    skyhloil.pbm
    <vacl> ---- low vacuum ----
    skyhvacl.pbm
    <vacr> ---- low vacuum ----
    skyhvacr.pbm
    <lowv> ---- low vacuum ----
    skyhlowv.pbm
    <volt> ---- volts ----
    skyhvolt.pbm
    <line> ---- warning line ----
    skyhwlin.pbm
    <lite> ---- lights ----
    anlt
    <help> ---- popup help ----
    Annunciator Panel
<endo> ==== END GAUGE ENTRY ====
```



shet Skyhawk Elevator Trim

Class: CSkyhawkElevatorTrim

Base Class: CBitmapGauge

Description:

This gauge type implements a specialized elevator trim wheel for the TRI Flyhawk aircraft.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK01.PNL)

```
<gage> ==== GAUGE ENTRY ====
shet
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    shet
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
    etrm
    <usr1> ---- user data ----
    101
  <endo>
  <size> ---- x,y,xsize,ysize ----
  373
  64
  79
  262
  <trmw> ---- trim wheel bitmap ----
  skyhtrim.pbg
  39
  3
  <trmi> ---- trim indicator bitmap ----
  skyhtrmi.pbg
  12
  89
  <help> ---- popup help ----
  Elevator Trim
  <down> ---- trim up click area ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
    0
    0
    79
    131
    <curs> ---- cursor ----
    mpflipup.csr
  <help> ---- popup help ----
  Trim Nose Down
  <endo> ---- end click area ----
  <up > ---- trim down click area ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
    0
    132
    79
    262
    <curs> ---- cursor ----
```



```
mpflipdn.csr  
  <help> ---- popup help ----  
    Trim Nose Up  
  <endo> ---- end click area ----  
<endo> ==== END GAUGE ENTRY ====
```

shfs Skyhawk Fuel Selector Switch

Class: CSkyhawkFuelSelector

Base Class: CBitmapGauge

Description:

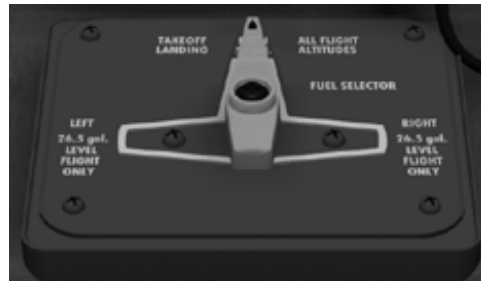
This gauge type implements a fuel tank selector switch for the TRI Flyhawk aircraft.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK01.PNL)

```
<gage> ==== GAUGE ENTRY ====
shfs
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    shfs
  <size> ---- x,y,xsize,ysize ----
    420
    434
    179
    82
  <swit> ---- fuel switch bitmap ---
    skyhfsel.pbg
  <help> ---- popup help ----
    Fuel Selector Switch
  <left> ---- left tank click area ----
  <bgn> ---- begin click area ----
    <rect> ---- click area rect ----
      0
      40
      70
      82
    <curs> ---- cursor ----
      mpfinger.csr
    <help> ---- popup help ----
      Left Tank
  <end> ---- end click area ----
  <both> ---- both tanks click area ----
  <bgn> ---- begin click area ----
    <rect> ---- click area rect ----
      55
      0
      125
      39
    <curs> ---- cursor ----
      mpfinger.csr
    <help> ---- popup help ----
      Both Tanks
  <end> ---- end click area ----
  <right> ---- right tank click area ----
  <bgn> ---- begin click area ----
    <rect> ---- click area rect ----
      109
      40
      179
      82
    <curs> ---- cursor ----
      mpfinger.csr
```



```
        <help> ---- popup help ----
        Right Tank
<endo> ---- end click area ----
<Lmsg> -- Left Tank Fuel Valve --
<bgn>
        <conn>
        Lone
        <dtag>
        st8t
<endo>
<Rmsg> -- Right Tank Fuel Valve --
<bgn>
        <conn>
        Rone
        <dtag>
        st8t
<endo>
<endo> ==== END GAUGE ENTRY ====
```

shlk Flyhawk Light Knob

Class: CFlyhawkLightKnobGauge

Base Class: CDualKnob

Description:

This specialized version of the dual knob gauge type is used to implement the panel lighting controls in the TRI Flyhawk aircraft. Each knob has an inner and an outer knob which can be manipulated independently.

This gauge type uses deprecated tags. Fly! 2 aircraft should not use this gauge type; use the generic dual knob type 'dknb' instead.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
shlk
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    rplk
  <usri> ---- user data ----
    HARDWARE,STATE
  <usri> ---- user data ----
    DATATAG,'lev1'
  <igrp> ---- user data ----
    rdlt
  <usro> ---- user data ----
    HARDWARE,STATE
  <usro> ---- user data ----
    DATATAG,'lev1'
  <ogrp> ---- user data ----
    gglr
  <size> ---- x,y,xsize,ysize ----
    584
    573
    42
    46
  <iinc> ---- inner knob increment ----
    0.05
  <idec> ---- inner knob decrement ----
    0.05
  <oinc> ---- outer knob increment ----
    0.05
  <odec> ---- outer knob decrement ----
    0.05
  <inca> ---- inner click area ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
      0
      0
      42
      23
    <cur> ---- cursor ----
    mpfinger.csr
  <help> ---- popup help ----
    Radio Lights
```



```
<endo> ---- end click area ----
<ouca> ---- outer click area ----
<bgno> ---- begin click area ----
      <rect> ---- click area rect ----
      0
      24
      42
      46
      <curs> ---- cursor ----
      mpfinger.csr
      <help> ---- popup help ----
      Panel Lights
<endo> ---- end click area ----
<help> ---- popup help ----
Radio/Panel Lights
<endo> ==== END GAUGE ENTRY ====
```

shts Skyhawk Annunciator Test Switch

Class: CSkyhawkAnnunciatorTestSwitch

Base Class: CBitmapGauge

Description:

This specific gauge implements the annunciator test switch for the TRI Flyhawk aircraft.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
shts
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    ants
  <swit> ---- switch artwork ----
    skyhtest.pbg
  <size> ---- x,y,xsize,ysize ----
    496
    127
    31
    34
  <csru> ---- up cursor ----
    mpflipup.csr
  <csrd> ---- down cursor ----
    mpflipdn.csr
  <sfxu> ---- up sfx ----
    switchup.wav
  <sfxd> ---- down sfx ----
    switchdn.wav
  <mesg> ---- message ----
  <bgno> ---- begin ----
    <conn> ---- connection ----
      anl1
    <user> ---- hardware ----
      HARDWARE,STATE
    <dtag> ---- data tag ----
      lev1
  <endo> ---- end ----
  <help> ---- popup help ----
    Annunciator Switch
<endo> ==== END GAUGE ENTRY ====
```



sios Simple In/Out Switch

Class: CSimpleInOutStateSwitch

Base Class: CBitmapGauge

Description:

Use this gauge type for knobs, switches or buttons that have only two states. Excellent for circuit breakers and pull out type knobs.

Read Tags

Signature	Description	Type	Format
sbmp	Switch bitmap	ART	Switch art (PBG)
stat	Default state	INT	
onvl on_ in_	"ON" value of switch (Use this to change the default state value)	INT	Default = 1
ofvl off_ out_	"OFF" value of switch (Use this to change the default state value)	INT	Default = 0
onht	"ON" state help text	WORD	
ofht	"OFF" state help text	WORD	
onfx infx	"ON" state sound effect	FILE	Sound effect (WAV)
ofsf oufx	"OFF" state sound effect	FILE	Sound effect (WAV)
mmnt	Momentary on	NO_VALUE	

Example (FLYHWK01.PNL)

```
<gage> ==== GAUGE ENTRY ====
sios
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    fcut
  <size> ---- x,y,xsize,ysize ----
    552
    365
    48
    72
  <mesg>
  <bgno>
    <conn>
    f0ff
    <dtag>
    st8t
  <endo>
  <onvl> ---- on value ----
    0
  <onsf> ---- on sfx ----
    pushknob.wav
  <ofvl> ---- off value ----
    1
  <ofsf> ---- off sfx ----
    pushknob.wav
  <sbmp> ---- bitmap name ----
    skyhfcut.pbg
  <crsr> ---- cursor ----
```



```
mpfinger.csr  
<help> ---- popup help ----  
Fuel Cutoff  
<endo> ==== END GAUGE ENTRY ====
```

slip Slip Indicator Gauge

Class: CSlipIndicatorGauge

Base Class: CBitmapGauge

Description:

This gauge type implements a standalone slip indicator. The artwork for the ball is specified by the <ball> tag.

Read Tags

Signature	Description	Type	Format
ball	Ball artwork	ART	PBG filename

Example (TRI30.PNL)

```
<gage> ==== Slip Ball ====
slip
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    slp1
    <size> ---- x,y,xsize,ysize ----
    317
    663
    34
    34
    <mesg> ---- message ----
    <bgno> ---- begin ----
        <conn>
        trn1
        <dtag>
        ball
    <endo> ---- end ----
    <ball> ---- ball bitmap ----
    kaahslip.pbg
    <help> ---- popup help ----
    Slip Indicator
<endo> ==== END GAUGE ENTRY ====
```



sngp Skyhawk Nav/Gps Switch

Class: CSkyhawkNavGpsPanel

Base Class: CBitmapGauge

This gauge type implements the NAV/GPS selector switch for the TRI Skyhawk aircraft.

Read Tags

Signature	Description	Type	Format
cnav	NAV Switch	OBJECT	Knob
capr	GPS APR Switch	OBJECT	Knob
xnav	NAV Indicator Position	INT	X pixel offset
		INT	Y pixel offset
xgps	GPS Indicator Position	INT	X pixel offset
		INT	Y pixel offset
xapr	APR Indicator Position	INT	X pixel offset
		INT	Y pixel offset
xwpt	WPT Indicator Position	INT	X pixel offset
		INT	Y pixel offset

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
sngp
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    sngp
  <size> ---- x,y,xsize,ysize ----
    556
    130
    77
    30
  <cnav> ---- NAV switch ----
  <bgn> ---- begin click area ----
    <rect> ---- click area rect ----
      23
      10
      36
      24
    <bmap> ---- bitmap ----
      csnavgps.pbg
    <curs> ---- cursor ----
      mpfinger.csr
    <help> ---- popup help ----
      NAV/GPS Slave
  <endo> ---- end click area ----
  <capr> ---- GPS APR switch ----
  <bgn> ---- begin click area ----
    <rect> ---- click area rect ----
      40
      10
      53
      24
    <bmap> ---- bitmap ----
      csgpsapr.pbg
```



```

        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        GPS Approach
    <endo> ---- end click area ----
    <xnav> ---- x,y NAV offset ----
    7
    10
    <xgps> ---- x,y GPS offset ----
    7
    18
    <xapr> ---- x,y APR offset ----
    55
    10
    <xmsg> ---- x,y MSG offset ----
    7
    3
    <xwpt> ---- x,y WPT offset ----
    55
    3
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- data tag ----
        K140
        <user> ---- user data ----
        HARDWARE, OTHER
    <endo>
    <help> ---- popup help ----
    NAV/GPS Selector Panel
<endo> ===== END GAUGE ENTRY =====

```

suct Suction Gauge

Class: CSuctionGauge

Base Class: CNeedleGauge

Description:

Unknown? How is this different from the 'vacu' gauge type?

Read Tags

Signature	Description	Type	Format

Example

swit Simple Switch

Class: CSimpleSwitchGauge

Base Class: CGauge

Description:

This common gauge type implements a simple multiple-position, discrete switch. Typical uses are for two- or three-position toggle switches.

Read Tags

Signature	Description	Type	Format
bmap	Switch bitmap	ART	Switch art (PBG)
sfxi	Sound effect for "in" or "down" position	FILE	Sound effect (WAV)
sfxo	Sound effect for "out" or "up" position	FILE	Sound effect (WAV)
stat	Switch states	INT	Number of states
		INT	First state value
		INT	Second state value
			... continue for number of states specified
sstr	State strings	INT	Number of states
		WORD	First state description
		WORD	Second state description
		...	continue for number of states specified
mmnt	Momentary contact	NO_VALUE	

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
swit
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  avio
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- data tag ----
    avio
    <user> ---- user data ----
    HARDWARE, SWITCH
  <endo>
  <size> ---- x,y,xsize,ysize ----
  343
  576
  33
  49
  <bmap> ---- bitmap name ----
  skyhavpw.pbg
  <csru> ---- up cursor ----
  mpflipup.csr
  <csrd> ---- down cursor ----
  mpflipdn.csr
  <sfxu> ---- up sfx ----
  switchup.wav
  <sfxd> ---- down sfx ----
  switchdn.wav
  <help> ---- popup help ----
```



Avionics Power
<endo> ==== END GAUGE ENTRY =====

sync Propeller Synchrophase Gauge

Class: CPropSyncFanGauge

Base Class: CBitmapGauge

Description:

This gauge type implements the propeller synchrophase indicator. There are no gauge-specific tags.

Read Tags

Signature	Description	Type	Format

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
sync
<bgn> ==== BEGIN GAUGE ====
  <unid> ---- unique id ----
  sync
  <size> ---- placement and size ----
  493
  878
  24
  24
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- info tag ----
    sync
    <user> ---- user data ----
    HARDWARE, GAUGE
  <endo>
  <bmap> ---- bitmap ----
  kapsw.pbg
  0
  0
  <help> ---- help string ----
  Synchrophaser
<endo> ==== END GAUGE =====
```



tach Tachometer Needle Gauge

Class: CTachometerGauge

Base Class: CNeedleGauge

Description:

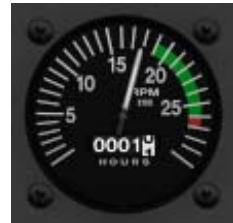
This specialized type of needle gauge implements an engine tachometer gauge which includes an integrated Hobbs meter.

Read Tags

Signature	Description	Type	Format
hobb	Hobbs Meter	OBJECT	Hobbs Meter

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
tach
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    rpms
  <size> ---- placement and size ----
    427
    410
    90
    90
  <mesg> -- connect to Tachometer Subsystem --
  <bgno>
    <conn>
      tach
    <dtag>
      rpms
  <endo>
  <cntr> ---- needle center x,y ----
    45
    46
  <nedl> ---- bitmap needle ----
  std
  <bgno> ==== BEGIN BITMAP NEEDLE ====
    <file> ---- needle file ----
      NEEDLE1.NDL
  <endo> ==== END BITMAP NEEDLE ====
  <gtbl> ---- gauge value table ----
  <bgno> ---- begin table ----
    <fmt1> - x y
    0.0 0.0
    200.0 14.3
    500.0 34.9
    700.0 57.8
    1000.0 72.8
    1200.0 88.7
    1500.0 111.8
    1700.0 129.5
    1900.0 148.4
    2000.0 158.1
    2100.0 167.2
    2200.0 176.0
    2300.0 185.1
    2400.0 195.2
    2500.0 204.8
    2700.0 223.3
    3000.0 250.7
  <endf> - end format
```



```

<endo> ---- end table ----
<sang> ---- starting angle ----
236.5
<gmin> ---- min value allowed ----
0.0
<gmax> ---- max value allowed ----
3000.0
<help> ---- popup help ----
Tachometer
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhrpmm.raw
<hobb> ---- HOBBS METER ----
<bgno> ---- BEGIN HOBBS ----
    <unid> ---- unique id ----
    hob1
    <size> ---- placement and size ----
    457
    465
    30
    9
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- info tag ----
        hobs
        <user> ---- user data ----
        HARDWARE, GAUGE
    <endo>
    <blak> ---- black digits ----
    skyhhmsb.pbg
    <whit> ---- white digits ----
    skyhhmsw.pbm
    <endo> ---- END HOBBS ----
<endo> ===== END GAUGE ENTRY =====

```

tcas TCAS Gauge

Class: CTCASGauge

Base Class: CGauge

Description:
Unknown?

Read Tags

Signature	Description	Type	Format

Example

tgrp Transponder Radio Group

Class: CTransponderRadioGroup

Base Class: CGauge

Description:
Unknown?

Read Tags

Signature	Description	Type	Format

Example

tknb Turn Knob

Class: CTurnKnobGauge

Base Class: CGauge

Description:

This gauge type is used to implement a rotatable knob that is used to control a float-type value. Typical examples include light brightness knobs.

Read Tags

Signature	Description	Type	Format
bmap	Knob artwork	ART	PBG filename
incr right	Increment amount	REAL	Amount value increases for each click Default = 0.1
decr left	Decrement amount	REAL	Amount value decreases for each click Default = 0.1

Example (TRI210.PNL)

```
<gage> ==== GAUGE ENTRY ====
tknb
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  plit
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
    plit
    <user> ---- user data ----
    HARDWARE,STATE
    <dtag> ---- user data ----
    levl
  <endo>
  <bmap> ---- bitmaps ----
  pnkblite.pbg
  <size> ---- x,y,xsize,ysize ----
  419
  350
  42
  41
  <curs> ---- cursor ----
  mpfinger.csr
  <help> ---- popup help ----
  Pilot Lights
<endo> ==== END GAUGE ENTRY ====
```



tswi Turn Switch

Class: CTurnSwitchGauge

Base Class: CGauge

Description:

This gauge type implements a rotatable multi-position discrete switch.

Read Tags

Signature	Description	Type	Format
bmap	Switch artwork	ART	PBG filename
stat	Switch states	INT	Number of states
		INT	First state value
		INT	Second state value
		...	...continue for number of state specified
sstr	State strings	INT	Number of states
		WORD	First state description
		WORD	Second state description
		...	...continue for number of states specified
sfxl	Sound effect for moving left	FILE	WAV filename
sfxr	Sound effect for moving right	FILE	WAV filename

Example (TRI200.PNL)

```
<gage> ==== gauge entry ====
tswi
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
    ackb
  <size> ---- x,y,xsize,ysize ----
    913
    588
    63
    62
  <bmap> ---- bitmap ----
    pnacknob.pbg
  < curs> ---- cursor ----
    mpfinger.csr
  <stat> ---- swith states -----
    7
    0
    1
    2
    3
    4
    5
    6
  <sstr> ---- state text ----
    7
    80
    77
    74
    72
    70
    68
```



```
65
  <help> ---- popup help ----
  Air Conditioner
<endo> ==== END GAUGE ENTRY ====
```

turn Turn Coordinator Gauge

Class: CTurnCoordinatorGauge

Base Class: CBitmapGauge

Description:

Read Tags

Signature	Description	Type	Format
plan	Plane plate artwork	ART	PBG filename
poff	Plane plate offset	INT	X pixel offset
		INT	Y pixel offset
ball	Ball artwork	ART	PBG filename
boff	Ball offset	INT	X pixel offset
		INT	Y pixel offset
pcon	Plane message data tag	ID_TYPE	datatag
bcon	Ball message data tag	ID_TYPE	datatag
Ltrn	Frame number of plane when it's in a left 2-minute turn	INT	Frame number
Rtrn	Frame number of plane when it's in a right 2-minute		

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
turn
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  turn
  <size> ---- x,y,xsize,ysize ----
  198
  287
  90
  90
  <mesg>
  <bgn>
    <conn>
    turn
    <dtag>
    turn
  <endo>
  <cntr> ---- needle center x,y ----
  45
  45
  <over> ---- overlay image ----
  turnolay.pbm
  2
  5
  <plan> ---- plane plate ----
  turncoor.pbg
  <poff> ---- plane offset ----
  2
  5
  <ball> ---- ball plate ----
  turnball.pbg
  <boff> ---- ball offset ----
  2
```



```
7
<pcon> ---- plane data tag ----
turn
<bcon> ---- ball data tag ----
ball
<help> ---- popup help ----
Turn Coordinator
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhtrnm.raw
<endo> ==== END GAUGE ENTRY ====
```

twhi Trim Wheel Indicator Gauge

Class: CTrimWheelIndicatorGauge

Base Class: CTrimWheelGauge

Description:

Read Tags

Signature	Description	Type	Format
<up >	Trim Up Click Area	OBJECT	Click Area
down	Trim Down Click Area	OBJECT	Click Area

Example (TRI31.PNL)

```
<gage> ==== GAUGE ENTRY ====
twhi
<bgno> ==== BEGIN OBJECT ====
  <unid>
  etrm
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> -- connection --
    etrm
    <dtag> ---- data tag ----
    unit
  <endo>
  <size> ---- size and pos ----
  90
  176
  108
  470
  <bmap> ---- bitmaps ----
  kaelvtrm.pbg
  <up > ---- trim up click area ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
    0
    0
    108
    235
    <curs> ---- cursor ----
    mpflipup.csr
    <help> ---- popup help ----
    Trim Nose Down
  <endo> ---- end click area ----
  <down> ---- trim down click area ----
  <bgno> ---- begin click area ----
    <rect> ---- click area rect ----
    0
    236
    108
    470
    <curs> ---- cursor ----
    mpflipdn.csr
    <help> ---- popup help ----
    Trim Nose Up
  <endo> ---- end click area ----
  <help> ---- popup help ----
  Elevator Trim
<endo> ==== END OBJECT =====
```



twhl Trim Wheel Control

Class: CTrimWheelGauge

Base Class: CGauge

Read Tags

Signature	Description	Type	Format

Example (TRI220.PNL)

```
<gage> ==== GAUGE ENTRY ====
twhl
<bgno> ==== BEGIN OBJECT ====
    <unid> ---- Unit ID ----
    etrm
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- Connecion ----
        etrm
        <dtag> ---- Data Tag ----
        unit
    <endo>
    <size> ---- size and pos ----
    318
    96
    66
    265
    <bmap> ---- bitmaps ----
    pneltrim.pbg
    <down> ---- trim up click area ----
    <bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        0
        0
        66
        132
        <curs> ---- cursor ----
        mpflipup.csr
        <help> ---- popup help ----
        Trim Nose Down
    <endo> ---- end click area ----
    <up > ---- trim down click area ----
    <bgno> ---- begin click area ----
        <rect> ---- click area rect ----
        0
        133
        66
        265
        <curs> ---- cursor ----
        mpflipdn.csr
        <help> ---- popup help ----
        Trim Nose Up
    <endo> ---- end click area ----
    <help> ---- popup help ----
    Elevator Trim
<endo> ==== END OBJECT =====
```

ufms Universal FMS

Class: CUniversalFMSGauge

Base Class: CBitmapGauge

Description:

This specialized gauge type implements the Universal Flight Management System.

Read Tags

Signature	Description	Type	Format
scrn	Display Screen	OBJECT	Click Area
l1__	Left Select Key 1	OBJECT	Click Area
l2__	Left Select Key 2	OBJECT	Click Area
l3__	Left Select Key 3	OBJECT	Click Area
l4__	Left Select Key 4	OBJECT	Click Area
l5__	Left Select Key 5	OBJECT	Click Area
r1__	Right Select Key 1	OBJECT	Click Area
r2__	Right Select Key 2	OBJECT	Click Area
r3__	Right Select Key 3	OBJECT	Click Area
r4__	Right Select Key 4	OBJECT	Click Area
r5__	Right Select Key 5	OBJECT	Click Area
data	DATA Button	OBJECT	Click Area
nav_	NAV Button	OBJECT	Click Area
vnav	VNAV Button	OBJECT	Click Area
dto_	DTO Button	OBJECT	Click Area
list	LIST Button	OBJECT	Click Area
prev	PREV Key	OBJECT	Click Area
fuel	PREV Key	OBJECT	Click Area
fpl_	PREV Key	OBJECT	Click Area
perf	PREV Key	OBJECT	Click Area
tune	TUNE Key	OBJECT	Click Area
menu	MENU Key	OBJECT	Click Area
next	NEXT Key	OBJECT	Click Area
1__	1 Key	OBJECT	Click Area
2__	2 Key	OBJECT	Click Area
3__	3 Key	OBJECT	Click Area
4__	4 Key	OBJECT	Click Area
5__	5 Key	OBJECT	Click Area
6__	6 Key	OBJECT	Click Area
7__	7 Key	OBJECT	Click Area
8__	8 Key	OBJECT	Click Area
9__	9 Key	OBJECT	Click Area
back	BACK Key	OBJECT	Click Area
0__	0 Key	OBJECT	Click Area
msg_	MSG Key	OBJECT	Click Area
dim_	DIM Key	OBJECT	Click Area
+_-	+/- Key	OBJECT	Click Area
entr	ENT Key	OBJECT	Click Area
A__	A Key	OBJECT	Click Area
B__	B Key	OBJECT	Click Area
C__	C Key	OBJECT	Click Area
D__	D Key	OBJECT	Click Area
E__	E Key	OBJECT	Click Area

F__	F Key	OBJECT	Click Area
G__	G Key	OBJECT	Click Area
H__	H Key	OBJECT	Click Area
I__	I Key	OBJECT	Click Area
J__	J Key	OBJECT	Click Area
K__	K Key	OBJECT	Click Area
L__	L Key	OBJECT	Click Area
M__	M Key	OBJECT	Click Area
N__	N Key	OBJECT	Click Area
O__	O Key	OBJECT	Click Area
P__	P Key	OBJECT	Click Area
Q__	Q Key	OBJECT	Click Area
R__	R Key	OBJECT	Click Area
S__	S Key	OBJECT	Click Area
T__	T Key	OBJECT	Click Area
U__	U Key	OBJECT	Click Area
V__	V Key	OBJECT	Click Area
W__	W Key	OBJECT	Click Area
X__	X Key	OBJECT	Click Area
Y__	Y Key	OBJECT	Click Area
Z__	Z Key	OBJECT	Click Area

Example (TRI46.PNL)

Due to the extreme length and repetitive nature of this gauge specification, an example is not copied here. Examples can be found in the panel file for the TRI Peregrine 800TR FMS panel in file TRI46.PNL



vacu Vacuum Gauge

Class: CVaccumGauge

Base Class: CNeedleGauge

Description:

This specialized needle gauge is intended to display the vacuum system pressure. There are no gauge-specific tags.

Read Tags

Signature	Description	Type	Format

Example (FLYHWK00.PNL)

```
<gage> ==== GAUGE ENTRY ====
vacu
<bgno> ==== BEGIN GAUGE ENTRY ====
    <unid> ---- unique id ----
    suct
    <over> ---- overlay image ----
    skyhvac.pbm
    0
    0
    <size> ---- placement and size ----
    109
    375
    33
    66
    <cntr> ---- needle center x,y ----
    13
    32
    <mesg> ---- Subsystem Message ----
    <bgno>
        <conn> ---- data tag ----
        suct
        <user> ---- user data ----
        HARDWARE, GAUGE
        <dtag>
        indn
    <endo>
    <nedl> ---- bitmap needle ----
    smal
    <bgno> ==== BEGIN BITMAP NEEDLE ====
        <file> ---- needle file ----
        needle3.ndl
    <endo> ==== END BITMAP NEEDLE ====
    <gtbl> ---- gauge value table ----
    <bgno> ---- begin table ----
        <fmt1> - x y
        0.0 142.8
        4.0 116.9
        5.0 90.0
        6.0 64.1
        7.0 38.4
        <endf> - end format
    <endo> ---- end table ----
    <gmin> ---- min value allowed ----
    0.0
    <gmax> ---- max value allowed ----
    7.0
    <help> ---- popup help ----
```

```
Vacuum Indicator
<lite> ---- light flag ----
gage
<mask> ---- light mask ----
skyhvacm.raw
<endo> ==== END GAUGE ENTRY ====
```

vert Autopilot VS Gauge

Class: CVerticalRateAltGauge

Base Class: CBitmapGauge

Description:

This gauge allows entry of a target vertical speed for the autopilot system.

Read Tags

Signature	Description	Type	Format
eng_	Engage Button	OBJECT	Knob
arm_	Arm Button	OBJECT	Knob
knob	VS Knob	OBJECT	Knob
led_	LED Display Area	OBJECT	Rectangle

Example (MALI00.PNL)

```
<gage> ==== GAUGE ENTRY ====
vert
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  vert
  <mesg> ---- Subsystem Message ----
  <bgn>
    <conn> ---- data tag ----
    AXIS
    <user> ---- user data ----
    HARDWARE,OTHER
    <dtag> ---- user data ----
    _Alt
  <endo>
  <size> ---- x,y,xsize,ysize ----
  512
  65
  107
  51
  <eng_> ---- Engage button ----
  <bgn> ---- begin button ----
    <rect> ---- rectangle ----
    3
    14
    13
    22
    <bmap> ---- bitmaps ----
    pmvseng.pbg
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    Engage Vertical Speed
  <endo> ---- end button
  <arm_> ---- Arm button ----
  <bgn> ---- begin button ----
    <rect> ---- rectangle ----
    3
    28
    13
    36
    <bmap> ---- bitmaps ----
    pmvsarm.pbg
    <curs> ---- cursor ----
```



```

        mpfinger.csr
        <help> ---- popup help ----
        Arm Altitude Select
<endo> ---- end button
<knob> ---- Rate knob ----
<bgn> ---- begin button ----
        <rect> ---- rectangle ----
        86
        16
        105
        34
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- popup help ----
        Rate Knob
<endo> ---- end button
<led_> ---- LED area ----
<bgn> ---- begin button ----
        <rect> ---- rectangle ----
        24
        14
        74
        39
<endo> ---- end button
<avil> ---- avionics light ----
        avio
        <help> ---- popup help ----
        Vertical Rate and Altitude
<endo> ===== END GAUGE ENTRY =====

```

vsld Vertical Slider

Class: CVerticalSliderGauge

Base Class: CGauge

Description:

Read Tags

Signature	Description	Type	Format

Example

vspd Vertical Speed Indicator Needle Gauge

Class: CVerticalSpeedGauge

Base Class: CNeedleGauge

Description:

This specialized needle gauge type implements a vertical speed indicator, with integrated vertical speed bug.

Read Tags

Signature	Description	Type	Format
vsbg	Vertical speed bug	ART	PBG filename
		INT	X pixel offset
		INT	Y pixel offset
vskb	Vertical speed bug knob	OBJECT	Knob
vmsg	Vertical speed bug message	OBJECT	Message

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
vspd
<bgn> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  vspd
  <size> ---- x,y,xsize,ysize ----
  423
  637
  90
  90
  <mesg> -- Subsystem Message --
  <bgn>
    <conn>
    vsi1
    <dtag>
    vsi_
  <endo>
  <cntr> ---- needle center x,y ----
  45
  45
  <sang> ---- starting angle ----
  101.8
  <gmin> ---- min value allowed ----
  -4000.0
  <gmax> ---- max value allowed ----
  4000.0
  <nedl> ---- bitmap needle ----
  ndl7
  <bgn> ==== BEGIN BITMAP NEEDLE ====
    <file> ---- needle file ----
    needle7.ndl
  <endo> ==== END BITMAP NEEDLE ====
  <gtbl> ---- gauge value table ----
  <bgn> ---- begin table ----
    <fmt1> - x y
    -4000.0 0.0
    -3000.0 26.0
    -2000.0 69.3
    -1000.0 123.2
    -500.0 148.7
```



```

        0.0 168.2
        500.0 186.9
        1000.0 212.7
        2000.0 267.4
        3000.0 310.8
        4000.0 336.8
        <endf> - end format
<endo> ---- end table ----
<vsbg> ---- VS bug ----
vsbug.pbg
5
5
<vskb> ---- VS knob ----
<bgno> ---- begin ----
    <rect> ---- rect ----
        1
        67
        15
        81
        <bmap> ---- bitmap ----
        kavsknob.pbg
        <help> ---- help ----
        Vertical Rate Preselect
<endo> ---- end ----
<vmmsg> ---- VS bug message ----
<bgno> ---- begin ----
    <conn> ---- group ----
        AXIS
        <user> ---- hardware ----
        HARDWARE, OTHER
        <dtag> ---- datatag ----
        VS_C
<endo> ---- end ----
<help> ---- popup help ----
Vertical Speed Indicator
<lite> ---- light flag ----
pfl1
<mask> ---- light mask ----
kamask08.raw
<endo> ===== END GAUGE ENTRY =====

```

wrad Weather Radar

Class: CWeatherRadarGauge

Base Class: CGauge

Description:

This gauge type implements the weather radar display.

Read Tags

Signature	Description	Type	Format
scrn	Display Screen	OBJECT	Rectangle
wx__	WX Button	OBJECT	Click Area
vp__	VP Button	OBJECT	Click Area
map_	MAP Button	OBJECT	Click Area
nav_	NAV Button	OBJECT	Click Area
rng+	RNG+ Button	OBJECT	Click Area
rng-	RNG- Button	OBJECT	Click Area
trk+	TRK+ Button	OBJECT	Click Area
trk-	TRK- Button	OBJECT	Click Area
mode	Mode Knob	OBJECT	Knob
tilt	Tilt Knob	OBJECT	Knob
gain	Gain Knob	OBJECT	Knob
brit	Brightness	OBJECT	Knob

Example (TRI200.PNL)

```
<gage> ==== GAUGE ENTRY ====
wrad
<bgno> ==== BEGIN GAUGE ENTRY ====
  <unid> ---- unique id ----
  wrdr
  <mesg> ---- Subsystem Message ----
  <bgno>
    <conn> ---- data tag ----
    wrdr
    <user> ---- user data ----
    HARDWARE,OTHER
  <endo>
  <size> ---- x,y,xsize,ysize ----
  748
  321
  190
  137
  <scrn> ---- visual screen ----
  <bgno> ---- begin ----
    <rect> ---- output rect ----
    35
    22
    151
    109
  <endo> ---- end ----
  <wx__> ---- WX button ----
  <bgno> ---- begin ----
    <rect> ---- output rect ----
    5
    33
    22
    45
    <curs> ---- cursor ----
    mpfinger.csr
```



```

        <help> ---- help ----
        WX button
<endo> ---- end ----
<vp__> ---- VP button ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        5
        51
        22
        63
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        VP button
<endo> ---- end ----
<map_> ---- MAP button ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        5
        69
        22
        81
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        MAP button
<endo> ---- end ----
<nav_> ---- NAV button ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        5
        87
        22
        99
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        NAV button
<endo> ---- end ----
<rng+> ---- RNG+ button ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        165
        33
        182
        45
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Increase Range
<endo> ---- end ----
<rng-> ---- RNG- button ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        165
        51
        182
        63
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Decrease Range button
<endo> ---- end ----
<trk+> ---- TRK+ button ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        165
        87
        182
        99
        <curs> ---- cursor ----

```

```

        mpfinger.csr
        <help> ---- help ----
        Track Right
<endo> ---- end ----
<trk-> ---- TRK- button ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        165
        69
        182
        81
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Track Left
<endo> ---- end ----
<mode> ---- Mode knob ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        157
        5
        188
        32
        <curs> ---- cursor ----
        mpfinger.csr
        <bmap> ---- bitmap ----
        bkwrmode.pbg
        <help> ---- help ----
        Power/Mode Knob
<endo> ---- end ----
<tilt> ---- TILT knob ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        164
        105
        184
        126
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Radar Tilt
<endo> ---- end ----
<gain> ---- GAIN knob ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        6
        109
        21
        124
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Radar Gain
<endo> ---- end ----
<brit> ---- BRT knob ----
<bgno> ---- begin ----
        <rect> ---- output rect ----
        6
        7
        21
        22
        <curs> ---- cursor ----
        mpfinger.csr
        <help> ---- help ----
        Brightness
<endo> ---- end ----
<lite> ---- light ----
        alit
        <help> ---- popup help ----
        Weather Radar
<endo> ==== END GAUGE ENTRY ====

```

xpdr Generic Transponder Gauge

Class: CGenericTransponderGauge

Base Class: CBitmapGauge

Description:

This gauge type implements a generic Transponder Radio. The <radi> tag specifies the ID and unit number of the associated radio subsystem.

The active and standby transponder codes are each displayed as clickable regions, with each digit being independently settable by clicking left/right on the digit. Click area objects for the active code are <ida1> through <ida4>. Click areas for the standby code are <idb1> through <idb5>.

If the <sdim> tag is included in the gauge specification, then the standby frequency is displayed in a dimmed font colour.

Read Tags

Signature	Description	Type	Format
ida1	Active Code Digit 1	OBJECT	Click Area
ida2	Active Code Digit 2	OBJECT	Click Area
ida3	Active Code Digit 3	OBJECT	Click Area
ida4	Active Code Digit 4	OBJECT	Click Area
idb1	Standby Code Digit 1	OBJECT	Click Area
idb2	Standby Code Digit 2	OBJECT	Click Area
idb3	Standby Code Digit 3	OBJECT	Click Area
idb4	Standby Code Digit 4	OBJECT	Click Area

Example (TRI30.PNL)

```
<gage> ==== GAUGE ENTRY ====
xpdr
<bgn> ==== BEGIN OBJECT ====
  <unid> ---- unique id ----
  xpdr
  <size> ---- x,y,xsize,ysize ----
  690
  802
  69
  71
  <radi> ---- radio identifier ----
  xpdr
  1
  <ida1> ---- ident 1 digit 1 ----
  <bgn> ---- begin click area ----
    <rect> ---- click area rect ----
    24
    5
    29
    17
    <curs> ---- cursor ----
    mpfinger.csr
  <endo> ---- end click area ----
  <ida2> ---- ident 1 digit 2 ----
  <bgn> ---- begin click area ----
    <rect> ---- click area rect ----
    30
```



```

5
35
17
<curs> ---- cursor ----
mpfinger.csr
<endo> ---- end click area ----
<ida3> ---- ident 1 digit 3 ----
<bgn0> ---- begin click area ----
<rect> ---- click area rect ----
36
5
41
17
<curs> ---- cursor ----
mpfinger.csr
<endo> ---- end click area ----
<ida4> ---- ident 1 digit 4 ----
<bgn0> ---- begin click area ----
<rect> ---- click area rect ----
42
5
47
17
<curs> ---- cursor ----
mpfinger.csr
<endo> ---- end click area ----
<idb1> ---- ident 2 digit 1 ----
<bgn0> ---- begin click area ----
<rect> ---- click area rect ----
24
18
29
30
<curs> ---- cursor ----
mpfinger.csr
<endo> ---- end click area ----
<idb2> ---- ident 2 digit 2 ----
<bgn0> ---- begin click area ----
<rect> ---- click area rect ----
30
18
35
30
<curs> ---- cursor ----
mpfinger.csr
<endo> ---- end click area ----
<idb3> ---- ident 2 digit 3 ----
<bgn0> ---- begin click area ----
<rect> ---- click area rect ----
36
18
41
30
<curs> ---- cursor ----
mpfinger.csr
<endo> ---- end click area ----
<idb4> ---- ident 2 digit 4 ----
<bgn0> ---- begin click area ----
<rect> ---- click area rect ----
42
18
47
30
<curs> ---- cursor ----
mpfinger.csr
<endo> ---- end click area ----
<tune> ---- tune click area ----
<bgn0> ---- begin click area ----
<rect> ---- click area rect ----
46
53

```

```

63
70
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Tune Knob
<endo> ---- end click area ----
<mode> ---- mode click area ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
8
48
31
73
<curs> ---- cursor ----
mpfinger.csr
<bmap> ---- bitmap ----
kaxpdrkb.pbg
<help> ---- popup help ----
Mode Knob
<endo> ---- end click area ----
<idnt> ---- Ident click area ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
32
47
43
58
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Ident Button
<endo> ---- end click area ----
<test> ---- Test click area ----
<bgno> ---- begin click area ----
<rect> ---- click area rect ----
32
59
43
70
<curs> ---- cursor ----
mpfinger.csr
<help> ---- popup help ----
Test Button
<endo> ---- end click area ----
<help> ---- popup help ----
Transponder Radio
<endo> ===== END OBJECT =====

```

Message Object

Description:

The Message object specifies a subsystem (defined in the aircraft .AMP file) that the gauge object exchanges messages with. In Fly! 2, the most important tags are the <conn> tag which specifies the unique subsystem identifier, and usually the <dtag> tag which specifies a specific data value that the subsystem supplies. In Fly!, it was possible to have non-unique subsystem identifiers so long as other tags (<unit>, <hwid>, etc.) could make the messaging target unique; this was changed in Fly! 2 to make the messaging simpler to understand, since all data values either GET-able or SET-able in a subsystem should be able to be addressed with just <conn> and <dtag>. Use of the old tags <user> and <type> is not recommended. The Subsystem Reference Guide contains a comprehensive list of all datatag values that can be retrieved or set for each subsystem type.

Tags:

Signature	Description	Type	Format
conn	Subsystem ID	ID_TYPE	
dtag	Datatag ID	ID_TYPE	
user	User Data	WORD,VALUE	Not recommended for use
type	Data Type	WORD,VALUE	Not recommended for use

Table Object

Description:

Tables are used to define specific mappings between variables. Typical uses of gauges in tables might be to map gauge indications to needle angles, etc.

Tables of format 1 (specified with the <fmt1> tag) consist of any number of x,y data pairs. See the separate Fly! 2 HowTo article on <fmt1> tables. Tables of format 3 (specified with the <fmt3> tag) consist of any number of polynomial coefficients

Tags:

Signature	Description	Type	Format
fmt1	Table Format 1	REAL REAL	First X Y data point
		REAL REAL	Second X Y data point
		...	etc. for all data points
fmt3	Table Format 3	REAL ...	Polynomial coefficients for first data point
		REAL ...	Polynomial coefficients for second data point
		...	etc. for all data points

Needle Object

Description:

The Needle object is a complex specification that is shared by most of the needle gauge types ('need', '2ned', '3ned' and their descendents).

Tags:

Signature	Description	Type	Format
file	Needle File	FILE	Specifies the name of a .NDL file to load needle attributes from. In most cases, you'll want to place all of the tags in a .NDL file so that when the gauge wants to specify a needle, it can simply specify the <file> tag without having to replicate the needle specifications in each and every gauge.
sizx	Needle X size	INT	Width of needle artwork (per frame) in pixels. By default, Fly! will automatically determine the width by reading the frame dimensions from the .PBG file. Only specify this tag if you wish to override the actual frame size.
sizey	Needle Y size	INT	Height of needle artwork (per frame) in pixels. By default, Fly! will automatically determine the width by reading the frame dimensions from the .PBG file. Only specify this tag if you wish to override the actual frame size.
offx	Center X offset	INT	X offset in pixels to the center (rotation point) of the needle.
offy	Center Y offset	INT	Y offset in pixels to the center (rotation point) of the needle.
base	Needle bitmap base name	STRING	Base filename (no PBG extension)
mind	Minimum degrees	REAL	Minimum displayable degrees. Use only if you have a needle that does not cover a 0..360 degree range.
maxd	Maximum degrees	REAL	Maximum displayable

			degrees. Use only if you have a needle that does not cover a 0..360 degree range.
shar	Shared Needle	NO_TYPE	Marks if this needle is shared across multiple aircraft. If so, the needle will remain memory resident even after the current aircraft the user is flying has been swapped. This helps reduce subsequent simulation load times.

Rectangle Object

Description:

The Rectangle object is used to define a simple rectangular area within the gauge.

Tags:

Signature	Description	Type	Format
rect	Rectangle Boundaries	INT	Top-left X pixel location
		INT	Top-left Y pixel location
		INT	Bottom-right X pixel location
		INT	Bottom-right Y pixel location

Example:

This example comes from the 'gslv' (Gyro Slaving Switch) gauge type:

```
<err_> ---- error window ----
<bgno> ---- begin button ----
    <rect> ---- rectangle ----
        14
        13
        45
        25
    <endo> ---- end button
```

Click Area Object

Description:

The Click Area object defines a rectangular area within the gauge that has some special function attached to it when the user clicks the mouse in the defined region. In addition to the <rect> tag which specifies the boundaries of the click area, custom pop-up help text can be specified with the <help> tag, and a custom cursor for use when the cursor is in the click area can be specified with the <curs> tag.

Tags:

Signature	Description	Type	Format
rect	Click Area Boundaries	INT	Top-left X pixel location
		INT	Top-left Y pixel location
		INT	Bottom-right X pixel location
		INT	Bottom-right Y pixel location
curs	Custom Cursor	CSR	CSR filename
help	Help Text	SENTENCE	

Example:

There are many examples of click areas throughout dozens of gauge types. This example shows the COM transfer button click area from the Bendix-King KX-155 NAV/COMM Radio (gauge type 'k155'):

```
<ca13> ---- COM transfer click area ----
<bgno> ---- begin click area ----
    <bmap> ---- bitmap ----
    skyhbt01.pbg
    <rect> ---- click area rect ----
    46
    39
    66
    53
    <curs> ---- cursor ----
    mpfinger.csr
    <help> ---- popup help ----
    COM Transfer Button
<endo> ---- end click area ----
```

Knob Object

Description:

The Knob object is a more advanced user interaction region within the gauge. In addition to the common click area tags, the Knob object also supports the specification of a knob bitmap animation with the <bmap> tag. The PBG file can contain any number of sub-images; as the knob is "turned" by left- and right-clicking on the region, the displayed frame is continuously decremented or incremented, respectively. The knob artwork should be designed so that one complete revolution is represented by all frames of the PBG file.

Tags:

Signature	Description	Type	Format
rect	Click Area Boundaries	INT	Top-left X pixel location
		INT	Top-left Y pixel location
		INT	Bottom-right X pixel location
		INT	Bottom-right Y pixel location
curs	Custom Cursor	CSR	CSR filename
help	Help Text	SENTENCE	
bmap	Knob artwork	ART	PBG filename

Example:

This example is the Mode Knob from the generic ADF gauge type 'adfr'. Note that while the comment says this is a click area, it is actually a Knob object.

```
<mode> ---- mode click area ----
<bgno> ---- begin click area ----
    <rect> ---- click area rect ----
        6
        48
        28
        72
    <curs> ---- cursor ----
        mpfinger.csr
    <bmap> ---- bitmap ----
        kaadfmod.pbg
    <help> ---- popup help ----
        Mode Knob
<endo> ---- end click area ----
```