

# Using Vega Prime Course Syllabus

---

Session duration: Classroom 4 days

## Main Objective

In this course, you will learn how to use LynX Prime to write an Application Configuration File (ACF) containing geometry, motion strategies, dynamic objects, light sources and environment effects.

Upon completion of this course, you will be able to write an application interacting with an ACF using the basic Vega Prime Graphical User Interface (GUI) and Application Program Interface (API).

## Target Audience

This is an ideal course for a new Vega Prime user, as it completely familiarizes you with the Vega Prime application building methodology.

## Prerequisites

For the first half: This course assumes basic PC knowledge.

For the second half: Exercises will be done in C++, so C++ programming is desirable.

## Format

This Instructor-led course is taught through a series of lectures and hands-on exercises in which you learn how to use all of the components of the tool.

## Topics Covered

- LynX Prime overview
- Entity management and movement
- vpPath module
- Collision detection
- Viewpoint management
- Aesthetics management
- vpLight
- Vega Prime API
- vpApp class and exercise
- vuBase
- vpTransform class and exercise
- vsNodes and vpGeometry
- vsDOF class and exercise
- vsSwitch class and exercise
- Subscribers
- vpIsector class and exercise
- vpChannel class and exercise
- Compiling your project

# Daily Outline for Classroom “Using Vega Prime” Course

## Day 1

- Lesson GUI-1: LynX Prime overview
- Lesson GUI-2: Entity management
- Lesson GUI-3: Entity movement
- Lesson GUI-4: vpPath module

## Day 2

- Lesson GUI-5: Collision detection
- Lesson GUI-6: Viewpoint management
- Lesson GUI-7: Aesthetics management
- Lesson GUI-8: vpLight

## Day 3

- Lesson API-1: Vega Prime API
- Lesson API-2: vpApp class
- Lesson API-3: vpApp exercise
- Lesson API-4: vuBase
- Lesson API-5: vpTransform class
- Lesson API-6: vpTransform exercise
- Lesson API-7: vsNodes and vpGeometry
- Lesson API-8: vsDOF class
- Lesson API-9: vsDOF exercise
- Lesson API-10: vsSwitch class
- Lesson API-11: vsSwitch exercise

## Day 4

- Lesson API-12: Subscribers
- Lesson API-13: vplsector class
- Lesson API-14: vplsector exercise
- Lesson API-15: vpChannel class
- Lesson API-16: vpChannel exercise
- Lesson API-17: Compiling your project
- Recap of the 25 lessons
- Questions and Answers specific from trainees
- Closing the course

# Detailed Description

## Lesson GUI-1: Vega Prime Overview

- The Big Picture

## Lesson GUI-2: LynX Prime overview

- Vega Prime system architecture
- Vega Prime modules
- Exploring the LynX Prime user interface
- EXERCISE 2-1: Look at the default ACF
- Preview the default ACF
- EXERCISE 2-2: Use Active Preview
- Using Selected Application

## Lesson GUI-3: Entity management

- Optimized 3D model features
- vpObject description
- EXERCISE 3-1: Create your own ACF
- EXERCISE 3-2: Create an F16
- vpTransform properties
- EXERCISE 3-3: Use a Transform
- vpScene definition
- vpObserver
- EXERCISE 3-4: Entity management

## Lesson GUI-4: Entity movement

- vpMotion Definition
- EXERCISE 4-1: vpMotionDrive
- vpMotionFly properties
- EXERCISE 4-2: vpMotionFly
- vpMotionGame properties
- vpMotionSpin properties
- vpMotionUFO properties
- vpMotionWalk properties
- vpMotionWarp properties
- vpMotionViewer properties
- EXERCISE 4-3: vpMotionViewer
- Tethering modes

- EXERCISE 4-4: Use different Tether mode
- EXERCISE 4-5: Drive
- EXERCISE 4-6: Fly
- EXERCISE 4-7: Tether Follow the hummer
- EXERCISE 4-8: Tether Spin the Thunderbird
- EXERCISE 4-9: Tether Fixed

#### **Lesson GUI-5: vpPath module**

- Classes of the Path module
- vpPathModule summary
- EXERCISE 5-1: Putting F16 on a Path
- EXERCISE 5-2: Thunderbird flying on a Path

#### **Lesson GUI-6: Collision detection**

- vplsector definition
- EXERCISE 6-1: Isector Tripod
- vplsector tripod properties
- EXERCISE 6-2: Isector Bump
- vplsector Line of Sight (LOS) properties
- EXERCISE 6-3: Isector Line of Sight
- vplsector Height Above Terrain (HAT) properties
- vplsector XYZPR properties
- vplsector ZPR properties
- vplsector Z properties
- EXERCISE 6-4: Tripod Isector with ground clamp
- EXERCISE 6-5: Line Of Sight Isector

#### **Lesson GUI-7: Viewpoint management**

- vpObserver
- vpChannel
- EXERCISE 7-1: Channel FOV
- EXERCISE 7-2: Two asymmetric channels
- More Channel Properties
- EXERCISE 7-3: Orthographic Channel
- EXERCISE 7-4: Create a Thunderbird View
- EXERCISE 7-5: Modifying the window

#### **Lesson GUI-8: Aesthetics management**

- vpEnvironment definition
- EXERCISE 8-1: Set Time of Day

- Celestial body properties
- vpShadowDynamicShadows definition
- EXERCISE 8-2: Shadows
- EXERCISE 8-3: Sun, Moon, Stars & Visibility
- Clouds Types
- EXERCISE 8-4: Cloud

#### **Lesson GUI-9: vpLight**

- vpLight definition
- EXERCISE 8-1: New light
- EXERCISE 8-2: Spotlight for the Hummer

#### **Lesson API-1: Vega Prime API**

- Vega Prime system architecture
- Runtime system control
- Instance management
- Best practices for Vega Prime real-time application development

#### **Lesson API-2: vpApp class**

- Methods of vpApp
- Inheritance - vpApp
- vp\_app\_minimal.cpp
- vp\_app\_basic.cpp
- Examples to review for vpApp
- EXERCISE 2-1: Hello World - Vega Prime style
- Running the application
- There is no Chapter 3

#### **Lesson API-4: vuBase**

- Methods of vuBase
- Inheritance
- vu\_base.cpp
- Other Examples

#### **Lesson API-5: vpTransform class**

- Methods of vpTransform
- Inheritance
- Examples to review for vpTransform

#### **Lesson API-6: vpTransform exercise**

- Transform Interactive Coding Exercises

- EXERCISE 6-1: Interactive Coding
- EXERCISE 6-2: Interactive Coding Fill the Blanks
- EXERCISE 6-3: Hands On Coding Lab1

#### **Lesson API-7: vsNodes and vpGeometry**

- vsNode class
- vpGeometry class
- Where did it all come from?
- Methods of vsNode
- Inheritance
- Methods of vpGeometry
- Creator nodes as interpreted by Vega Prime and VSG

#### **Lesson API-8: vsDOF class**

- Methods of vsDOF
- vs\_DOF.cpp
- About constraints
- vs\_DOF\_constraints.cpp
- Inheritance
- Examples to review for vsDOF

#### **Lesson API-9: vsDOF exercise**

- vsDOF class
- DOF Interactive Coding Exercises
- EXERCISE 9-1: Interactive Coding
- EXERCISE 9-2: Interactive Coding Fill the Blanks
- EXERCISE 9-3: Hands On Coding - Lab2

#### **Lesson API-10: vsSwitch class**

- Methods of vsSwitch
- vs\_switch.cpp
- Inheritance
- Examples to review for vsSwitch

#### **Lesson API-11: vsSwitch exercise**

- Switch Interactive Coding Exercises
- EXERCISE 11-1: Interactive Coding
- EXERCISE 11-2: Interactive Coding Fill the Blanks
- EXERCISE 11-3: Hands On Coding - Lab 3
- EXERCISE 11-4: Extra Credit

### **Lesson API-12: Subscribers**

- Subscriber class
- Subscriber methods
- Subscriber events for each class
- Examples to review for subscribers

### **Lesson API-13: vplsector class**

- vplsector methods
- Inheritance
- vpHAT class
- vplsectorHAT class methods
- vplsectorHAT.cpp
- vplsectorBump
- vplsectorBump methods
- vp\_isector\_bump.cpp
- vplsectorLOS
- vplsectorLOS methods
- vp\_isector\_LOS.cpp
- vplsectorTripod class
- vplsectorTripod methods
- vp\_isector\_tripod.cpp
- vplsectorXYZPR class
- vplsectorXYZPR methods
- vp\_isector\_XYZPR.cpp
- vplsectorZ methods
- vp\_isector\_Z.cpp
- vplsectorZPR class
- vplsectorZPR methods
- Inheritance
- vp\_isector\_ZPR.cpp
- Examples to review for vplsectors

### **Lesson API-14: vplsector exercise**

- Isector Interactive Coding Exercises
- EXERCISE 14-1: Interactive Coding
- EXERCISE 14-2: Interactive Coding Fill the Blanks
- EXERCISE 14-3: Hands On Coding - Lab4

### **Lesson API-15: vpChannel class**

- vpChannel methods
- Inheritance
- Examples to review for vpChannel

### **Lesson API-16: vpChannel exercise**

- Channel Interactive Coding Exercises
- EXERCISE 16-1: Interactive Coding
- EXERCISE 16-2: Interactive Coding Fill the Blanks
- EXERCISE 16-3: Hands On Coding - Lab 5

### **Lesson API-17: Compiling your project**

- Using Microsoft Visual C++ 2008 or 2010 with Vega Prime
- Environment variables
- Manually create your Visual C++ training solution and projects
- Compiling and running a Vega Prime application
- Manually Create your Visual C++ Training Solution and Projects
- To Compile Your Project
- To Execute Your Project